Adaptive Sampling for Minimax Fair Classification

Shubhanshu Shekhar [*] **Greg Fields** [*] **Mohammad Ghavamzadeh** [†] **Tara Javidi** [*]

Abstract

Machine learning models trained on uncurated datasets can often end up adversely affecting inputs belonging to underrepresented groups. To address this issue, we consider the problem of adaptively constructing training sets which allow us to learn classifiers that are fair in a *minimax* sense. We first propose an adaptive sampling algorithm based on the principle of *optimism*, and derive theoretical bounds on its performance. We also propose heuristic extensions of this algorithm suitable for application to large scale, practical problems. Next, by deriving algorithm independent lower-bounds for a specific class of problems, we show that the performance achieved by our adaptive scheme cannot be improved in general. We then validate the benefits of adaptively constructing training sets via experiments on synthetic tasks with logistic regression classifiers, as well as on several real-world tasks using convolutional neural networks (CNNs).

1 Introduction

Machine learning (ML) models are increasingly being applied for automating the decision-making process in several sensitive applications, such as loan approval and employment screening. However, recent work has demonstrated that discriminatory behaviour might get encoded in the model at various stages of the ML pipeline, such as data collection, labelling, feature selection, and training, and as a result, adversely impact members of some protected groups in rather subtle ways (Barocas and Selbst, 2016). This is why ML researchers have started to introduce a large number of *fairness measures* to include the notion of fairness in the design of their algorithms. Some of the important measures of fairness include demographic parity (Zemel et al., 2013), equal odds and opportunity (Hardt et al., 2016; Woodworth et al., 2017), individual fairness (Dwork et al., 2012), and minimax fairness (Feldman et al., 2015). The *minimax* fairness is particularly important in scenarios in which it is necessary to be as close as possible to equality without introducing unnecessary harm (Ustun et al., 2019). These scenarios are common in areas such as healthcare and predicting domestic violence. A measure that has been explored to achieve this goal is predictive risk disparity (Feldman et al., 2015; Chen et al., 2018; Ustun et al., 2019). Instead of using the common approach of putting constraints on the norm of discrimination gaps, Martinez et al. (2020) has recently introduced the notion of *minimax Pareto fairness*. These are minimax classifiers that are on the Pareto frontier of prediction risk, i.e., no decrease in the predictive risk of one group is possible without increasing the risk of another one.

In this paper, we are primarily interested in the notion of minimax fairness in terms of the predictive risk. However, instead of studying the training phase of the ML pipeline, our focus is on the data-collection stage, motivated by Jo and Gebru (2019) and Holstein et al. (2018). In particular, we study the following question: *given a finite sampling budget, how should a learner construct a training set consisting of elements from different protected groups in appropriate proportions to ensure that a classifier trained on this dataset achieves minimax fairness?*

Our work is motivated by the following scenario: suppose we have to learn a ML model for performing a task (e.g., loan approval) for inputs belonging to different groups based on protected attributes, such

[*] Electrical and Computer Engineering Department at UCSD ({shshekha,grfields,tjavidi}@ucsd.edu)

[†] Google Research (ghavamza@google.com)

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

as race or gender. Depending upon the distribution of input-label pairs from different groups, the given task may be statistically harder for some groups. Our goal is to ensure that the eventual ML model has optimal predictive accuracy for the worst-off group. We show that, under certain technical conditions, this results in a model with comparable predictive accuracy over all groups. One approach for this would be to train separate classifiers for each group. However, this is often not possible as the group-membership information may not be available at the deployment time, or it may be forbidden by law to explicitly use the protected characteristics as an input in the prediction process (Lipton et al., 2017, § 1). To ensure having a higher proportion of samples from the *harder* groups without knowing the identity of the hard or easy groups apriori, we consider this problem in an *active setting*, where a learner has to incrementally construct a training set by drawing samples one at a time (or a batch at a time) from different groups. Towards this goal, we propose and analyze an adaptive sampling scheme based on the principle of *optimism*, used in bandits literature (e.g., Auer et al. 2002), that detects the *harder* groups and populates the training set with more samples from them in an adaptive manner. We also wish to note that bias in ML has multi-faceted origins and that our work here addresses dataset construction and cannot account for bias introduced by model selection, the underlying data distribution, or other sources as discussed in Hooker (2021), Suresh and Guttag (2019). We also endeavor to ensure *minimax* fairness, but in some contexts another notion of fairness, such as those mentioned above, may be more appropriate or equitable. In general, application of our algorithm is not a guarantee that the resulting model is wholly without bias.

Our *main contributions* are: **1)** We first propose an optimistic adaptive sampling strategy, $\mathcal{A}_{\texttt{opt}}$, for training set construction in Section 3. This strategy is suited to smaller problems and admits theoretical guarantees. We then introduce a heuristic variant of $\mathcal{A}_{\texttt{opt}}$ in Section 4 that is more suitable to practical problems involving CNNs. **2)** We obtain upper bounds on the convergence rate of $\mathcal{A}_{\texttt{opt}}$, and show its minimax near-optimality by constructing a matching lower bound in Section 3.1. **3)** Finally, we demonstrate the benefits of our algorithm with empirical results on several synthetic and real-world datasets in Section 5.

Related Work. The closest work to ours is by Abernethy et al. (2020), where they propose an ϵ-greedy adaptive sampling strategy. They present theoretical analysis under somewhat restrictive assumptions and also empirically demonstrate the benefits of their strategy over some baselines. We describe their results in more detail in Appendix C.1, and employ the tools we develop to analyze our algorithm to perform a thorough analysis of the excess risk of their strategy under a much less restrictive set of assumptions and to show some necessary conditions on the value of their exploration parameter, ϵ. We find comparable empirical results for both algorithms given sufficient tuning of their respective hyperparameters, we report some of these results in Section 5 and compare the algorithms in Appendix C. In another related work, Anahideh et al. (2020) propose a fair adaptive sampling strategy that selects points based on a linear combination of model accuracy and fairness measures, and empirically study its performance. These results, however, do not obtain convergence rates of the excess risk of their respective methods and only offer implementations for small-scale datasets.

The above results study the problem of fair classification in an *active* setting and target the *data-collection* stage of the ML pipeline. There are also works that take a *passive* approach to this problem and focus of the *training* phase of the pipeline. Agarwal et al. (2018) design a scheme for learning fair binary classifiers with fairness metrics that can be written as linear constraints involving certain conditional moments. This class of fairness metrics, however, do not contain the minimax fairness measure. Diana et al. (2021) propose a method for constructing (randomized) classifiers for minimax fairness w.r.t. the empirical loss calculated on the given training set. Similarly, Martinez et al. (2020) derive an algorithm for learning a Pareto optimal minimax fair classifier, under the assumption that the learner has access to the true expected loss functions. Thus, the theoretical guarantees in Diana et al. (2021) and Martinez et al. (2020) hold under the assumption of large training sets. Our work, in contrast, constructs the training set incrementally (active setting) from scratch while carefully taking into account the effects of the finite sample size. Finally, we note that the data-collection strategies proposed in our paper can, in principle, be combined with the training methods presented in Martinez et al. (2020) and Diana et al. (2021) to further guarantee the (minimax) fairness of the resulting classifier. We leave the investigation of this approach for future work.

Besides data-collection and training, there have been studies in the fair ML literature on other aspects of the ML pipeline, such as pre-processing (Celis et al., 2020), learning feature representations (Zemel et al., 2013), post-processing (Hardt et al., 2016), and model documentation (Mitchell et al., 2018). We refer readers to a recent survey by Caton and Haas (2020) for detailed description of these results.

2 Problem Formulation

Consider a classification problem with the feature (input) space $\mathcal{X}$, label set $\mathcal{Y}$, and a protected attribute set $\mathcal{Z} = \{z_1, \ldots, z_m\}$. For any $z \in \mathcal{Z}$, we use $P_z(x, y)$ as a shorthand for $P_{XY|Z=z}(X = x, Y = y \mid Z = z)$ to denote the feature-label joint distribution given that the protected feature value is $Z = z$. We also use $\mathbb{E}_z[\cdot]$ as a shorthand for the expectation w.r.t. P_z. A (non-randomized) classifier is a mapping $f : \mathcal{X} \mapsto \mathcal{Y}$, which assigns a label $y \in \mathcal{Y}$ to every feature $x \in \mathcal{X}$. For a family of classifiers $\mathcal{F} \subset \mathcal{Y}^{\mathcal{X}}$, a loss function $\ell : \mathcal{F} \times \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$, and a mixture distribution over the protected attributes $\pi \in \Delta_m$, we define the *π-optimal classifier* f_π as

$$f_\pi \in \operatorname*{arg\,min}_{f \in \mathcal{F}} \; \mathbb{E}_\pi \left[\ell(f, X, Y)\right] := \sum_{z \in \mathcal{Z}} \pi(z) \, \mathbb{E}_z \left[\ell(f, X, Y)\right]. \tag{1}$$

When π lies on the corners of the simplex Δ_m, i.e., $\pi(z) = 1$ for a $z \in \mathcal{Z}$, we use the notation f_z instead of f_π. For π in the interior of Δ_m, it is clear that the risk of f_π on group z, i.e., $\mathbb{E}_z \left[\ell(f_\pi, X, Y)\right]$, must be larger than the best possible classification loss for $P_{XY|Z=z}$. In this paper, our goal is to develop an *active sampling scheme* to find the fair mixture distribution π^* in a *minimax* sense (*minimizing the maximum risk among the groups*), i.e.,

$$\pi^* \in \operatorname*{arg\,min}_{\pi \in \Delta_m} \; \max_{z \in \mathcal{Z}} \; L(z, f_\pi) := \mathbb{E}_z \left[\ell(f_\pi, X, Y)\right]. \tag{2}$$

The active sampling problem that we study in this paper can be formally defined as follows:

Problem 1. *Suppose $\mathcal{F}$ denotes a family of classifiers, ℓ a loss function, n is a sampling budget, and $\mathcal{O} : \mathcal{Z} \mapsto \mathcal{X} \times \mathcal{Y}$ an oracle that maps any attribute $z \in \mathcal{Z}$ to a feature-label pair $(X, Y) \sim P_{XY|Z=z}$. The learner designs an adaptive sampling scheme $\mathcal{A}$ that comprises a sequence of mappings $(A_t)_{t=1}^n$ with $A_t : (\mathcal{X} \times \mathcal{Y})^{t-1} \mapsto \mathcal{Z}$ to adaptively query $\mathcal{O}$ and construct a dataset of size n. Let $\pi_n \in \Delta_m$ denote the resulting empirical mixture distribution over $\mathcal{Z}$, where $\pi_n(z) = N_{z,n}/n$ and $N_{z,n}$ is the number of times that $\mathcal{A}$ samples from P_z in n rounds. Then, the quality of the resulting dataset is measured by the excess risk, or sub-optimality, of the π_n-optimal classifier f_{π_n}, i.e.,*

$$\mathcal{R}_n(\mathcal{A}) := \max_{z \in \mathcal{Z}} \; L(z, f_{\pi_n}) \; - \; \max_{z \in \mathcal{Z}} \; L(z, f_{\pi^*}), \tag{3}$$

where π^* is the fair mixture distribution defined by (2)*. Hence, the goal of the learner is to design a strategy $\mathcal{A}$ which has a small excess risk $\mathcal{R}_n(\mathcal{A})$.*

Informally, the algorithm should adaptively identify the harder groups z and dedicate a larger portion of the overall budget n to sample from their distributions (see Section 2.2 for an illustrative example).

2.1 Properties of the Fair Mixture

As discussed above, our goal is to derive an active sampling scheme that allocates the overall budget n over the attributes $z \in \mathcal{Z}$ in a similar manner as the *unknown* fair mixture π^*, defined by (2). Thus, it is important to better understand the properties of π^* and the π-optimal classifiers f_π, defined by (1). We state three properties of f_π and π^* in this section. We refer the readers to Martinez et al. (2020) for the definitions of Pareto front and convexity discussed in this section.

Property 1. As discussed in Martinez et al. (2020, Section 4), any f_π that solves (1) for a π with $\pi(z) > 0, \; \forall z \in \mathcal{Z}$, belongs to the Pareto front $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$ of the risk functions $\{L(z, f)\}_{z \in \mathcal{Z}}, \; \forall f \in \mathcal{F}$.

Property 2. We prove in Proposition 1 (see Appendix A for the proof) that under the following two assumptions on the conditional distributions $\{P_z\}_{z \in \mathcal{Z}}$, function class $\mathcal{F}$, and loss function ℓ, there exists a unique fair mixture π^*, whose classifier f_{π^*} has equal risk over all attributes $z \in \mathcal{Z}$.

Assumption 1. *The mapping $\pi \mapsto L(z, f_\pi)$ is continuous for all attributes $z \in \mathcal{Z}$. Furthermore, if $\pi, \nu \in \Delta_m$ are such that $\pi(z) > \nu(z)$ for an attribute $z \in \mathcal{Z}$, then $L(z, f_\pi) < L(z, f_\nu)$.*

The above assumption indicates that increasing the weight of an attribute z in the mixture distribution must lead to an increase in the performance of the resulting classifier on the distribution P_z.

Assumption 2. *For any two distinct attributes $z, z' \in \mathcal{Z}$, we must have $L(z, f_z^*) < L(z', f_z^*)$, for any $f_z^* \in \arg\min_{f \in \mathcal{F}} L(z, f)$.*

This assumption requires that any optimal classifier corresponding to distribution P_z to have higher risk w.r.t. the distribution $P_{z'}$ of any other attribute $z' \in \mathcal{Z}$. Note that Assumption 2 may not always hold, for example, if one attribute is *significantly* easier to classify than another.

Proposition 1. *Let Assumptions 1 and 2 hold for the conditional distributions* $\{P_z\}_{z\in\mathcal{Z}}$*, the loss function* ℓ*, and the function class* $\mathcal{F}$*. Then, there exists a unique* $\pi^* \in \Delta_m$ *that achieves the optimal value for problem* (2) *and satisfies* $L(z_1, f_{\pi^*}) = L(z_2, f_{\pi^*}) = \cdots = L(z_m, f_{\pi^*}) := M^*$.

In other words, Assumptions 1 and 2 define a regime where there exists a fair mixture π^*, whose classifier f_{π^*} achieves complete parity in the classification performance across all attributes. Moreover, Property 1 indicates that f_{π^*} also belongs to the Pareto front $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$. Therefore, under these two assumptions, the equal risk classifier not only belongs to the Pareto front, but it is also the minimax Pareto fair classifier (Lemma 3.1 in Martinez et al. 2020). Note that, in general, without Assumption 1, the classifier that attains equality of risk might have worse performance on all attributes than the minimax Pareto fair classifier (Lemma 3.2 in Martinez et al. 2020).

Property 3. We may show that when both the function class $\mathcal{F}$ and risk functions $\{L(z,\cdot)\}_{z\in\mathcal{Z}}$ are *convex*, f_{π^*} is a minimax Pareto fair classifier. As originally derived by Geoffrion (1968) and then restated by Martinez et al. (2020, Theorem 4.1), under these convexity assumptions, the Pareto front $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$ is convex and any classifier on $\mathcal{P}_{\mathcal{Z},\mathcal{F}}$ is a solution to (1). This together with Property 1 indicates that the classifier f_{π^*} corresponding to the fair (minimax optimal) mixture π^* is on the Pareto front, and additionally, is a minimax Pareto fair classifier.

2.2 Synthetic Models

It is illustrative to consider a class of synthetic models which we use as a running example.

Definition 1 (`SyntheticModel1`)**.** Set $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0, 1\}$, and $\mathcal{Z} = \{u, v\}$. Each instance of our synthetic model is defined by the set of distributions $P_{XY|Z}$ that satisfy the following: $P_{Y|Z}(Y = 1|Z = z) = 0.5$ for both $z \in \mathcal{Z}$, and $P_{X|Y=y,Z=z} = \mathcal{N}(\mu_{yz}, I_2)$ for all $(y, z) \in \mathcal{Y} \times \mathcal{Z}$. Thus, each instance of this model can be represented by a mean vector $\boldsymbol{\mu} = (\mu_{yz} : y \in \mathcal{Y},\ z \in \mathcal{Z})$.

The idea behind this class of models is that the 'hardness' of the classification problem for a value of $z \in \mathcal{Z}$ depends on the distance between μ_{0z} and μ_{1z}. The more separated these two mean vectors are, the easier it is to distinguish the labels. Thus, depending on this distance, it is expected that different protected attributes may require different fractions of samples in the training set to achieve comparable test accuracy. To illustrate this, we consider two instances of the `SyntheticModel1`: **I.** $\mu_{0u} = (-2, 2)$, $\mu_{1u} = (2, -2)$, $\mu_{0v} = (-1, -1)$, and $\mu_{1v} = (1, 1)$, and **II.** $\mu_{0u} = (-1.5, 1.5)$, $\mu_{1u} = (1.5, -1.5)$, $\mu_{0v} = (-2, -2)$, and $\mu_{1v} = (2, 2)$.

For both model instances, we trained a logistic regression classifier over training sets with 1001 equally spaced values of $\pi(u)$ in $[0, 1]$. Figure 1 shows the test accuracy of the learned classifier for both attributes (blue and red curves) as well as their minimum (black curve) for different values of $\pi(u)$. Since the pair (μ_{0u}, μ_{1u}) is better separated than (μ_{0v}, μ_{1v}) in the first instance, it is easier to classify inputs with $Z = u$ than those with $Z = v$, and thus, it requires fewer training samples from $P_{z=u}$ than $P_{z=v}$ to achieve the same accuracy. This is reflected in Figure 1 *(left)* that shows the best (min-max) performance is achieved at $\pi(u) \approx 0.23$. An opposite trend is observed in the second instance (*right* plot in Figure 1), where the mean vectors corresponding to the attribute $Z = v$ are better separated, and hence, require fewer training samples to achieve the same accuracy.

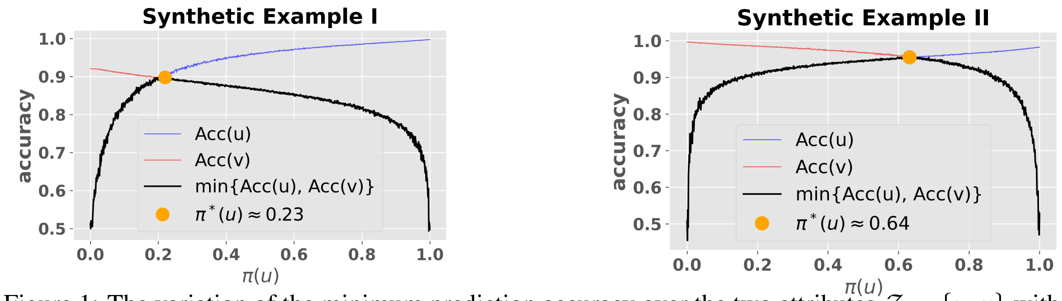

Figure 1: The variation of the minimum prediction accuracy over the two attributes $\mathcal{Z} = \{u, v\}$ with $\pi(u)$ for Instance I, on the left, and Instance II, on the right, of the `SyntheticModel1`.

3 Optimistic Adaptive Sampling Strategy

In this section, we present our *optimistic adaptive sampling* algorithm $\mathcal{A}_{\texttt{opt}}$ for Problem 1. Algorithm 1 contains a simple pseudo-code for $\mathcal{A}_{\texttt{opt}}$, see Algorithm 2 for a more detailed, formal pseudo-code. The algorithm proceeds in the following phases:

Phase 1. In the initial phase $t \leq m = |\mathcal{Z}|$, we draw two independent samples from $P_z = P_{XY|Z=z}$, for each attribute $z \in \mathcal{Z}$, and add one to the training dataset $\mathcal{D}_t$ and the other one to $\mathcal{D}_z$. We use $\mathcal{D}_t$ to learn a common (over all $z \in \mathcal{Z}$) classifier $\hat{f}_t$ via *empirical risk minimization* (ERM). The independent datasets $\{\mathcal{D}_{z_i}\}_{i=1}^m$ are used for estimating the performance of $\hat{f}_t$ for each $z \in \mathcal{Z}$.

Phase 2. In each round $t > m$, we choose an attribute z_t according to the following **selection rule**: *if* there exists an attribute z whose number of samples in $\mathcal{D}_t$, denoted by $N_{z_t,t}$, is fewer than t^ξ for some input $\xi \in (0,1)$ (Line 5), we set it as z_t (Line 6), *else* we set z_t as the attribute which has the largest upper confidence bound (UCB) (described below) for the risk of the classifier f_{π_t} (Line 8). Here π_t denote the empirical mixture distribution (over the attributes z) of $\mathcal{D}_t$.

Phase 3. When the attribute z_t is selected in *Phase 2*, the algorithm draws a pair of independent samples from P_{z_t}, adds one to $\mathcal{D}_t$ and the other to $\mathcal{D}_{z_t}$, and updates $N_{z_t,t}$, π_t, and the uniform deviation bound $e_{z_t}(N_{z_t,t})$ (described below) (Lines 10 to 12). The updated dataset $\mathcal{D}_t$ is then used to learn a new candidate classifier $\hat{f}_t$ (Line 13). *Phases 2 and 3* are repeated until the sampling budget is exhausted, i.e., $t = n/2$ (note that we sample twice at each round).

Calculating UCB. To construct the UCB, we introduce two additional assumptions:

For stating the next assumption, we will use the notations $L(z, f) = \mathbb{E}_z[\ell(f, X, Y)]$ and $L(\pi, f) = \sum_{z\in\mathcal{Z}} \pi(z)L(z, f)$ for any $f \in \mathcal{F}$, $z \in \mathcal{Z}$ and $\pi \in \Delta_m$.

Algorithm 1: Optimistic Adaptive Sampling for Minimax Fair Classification.

```
Input: $n$ (budget), $\mathcal{F}$ (function class),
       $\ell$ (loss function), $\xi$ (forced
       exploration term)
1  Initialize: $\mathcal{D}_1 = \emptyset$; $\{\mathcal{D}_{z_i}\}_{i=1}^m = \emptyset$;
       $\{N_{z_i,1}\}_{i=1}^m = 0$;
2  for $t = 1, \ldots, n/2$ do
3      if $t \leq m$ then
4          $z_t = z_t$;
5      else if $\min_{z\in\mathcal{Z}} N_{z,t} < t^\xi$ then
6          $z_t \in \arg\min_{z\in\mathcal{Z}} N_{z,t}$;
7      else
8          $z_t = \arg\max_{z\in\mathcal{Z}} U_t(z, \hat{f}_t)$;
               // see Equation 4
9      end
10     Draw two independent samples from
         $P_{z_t}$;
11     Add the first one to $\mathcal{D}_t$ and the second
         one to $\mathcal{D}_{z_t}$;
12     Update $N_{z_t,t}$; $\pi_t$; $e_{z_t}(N_{z_t,t})$;
13     Learn classifier $\hat{f}_t$ by ERM using $\mathcal{D}_t$;
14 end
Output: $\pi_{n/2}$ and $\hat{f}_{n/2}$
```

Assumption 3. *There exist positive constants $\epsilon_0, C > 0$ such that for any function $f \in \mathcal{F}$ and any $\pi \in \Delta_m$, with $\pi(z) > 0$, $\forall z \in \mathcal{Z}$, if we have $L(\pi, f) \leq L(\pi, f_\pi) + \epsilon$, for some $0 < \epsilon \leq \epsilon_0$, then $\pi(z) \times |L(z, f) - L(z, f_\pi)| \leq 2C\epsilon$, $\forall z \in \mathcal{Z}$.*

Assumption 3 is stated in abstract terms, so Example 1 describes a problem for which it holds.

Example 1. *Consider the binary classification problem, with $\mathcal{X} = [0,1]^k$ and $\mathcal{Y} = \{0,1\}$ and $\mathcal{Z} = \{z_1, \ldots, z_m\}$. Let $\mathcal{F}$ be the given family of classifiers $f : \mathcal{X} \to \mathcal{Y}$, and introduce $d(f_1, f_2) = \mu(\{f_1 \neq f_2\})$, where μ is the Lebesgue measure on $\mathcal{X}$. Assume that the marginal distributions $P_{X|Z=z}$ for all $z \in \mathcal{Z}$ admit densities ν_z with respect to μ satisfying $C_1 \leq \nu_z(x) \leq C_2$ for all $x \in \mathcal{X}$. Note that this is a version of the* strong-density *assumption used in prior works in classification, such as (Audibert and Tsybakov, 2007, Definition 2.2). Then, for the ℓ_{01} loss, we observe that the expected loss between any two f_1, f_2 satisfies $C_1 d(f_1, f_2) \leq |L(z, f_1) - L(z, f_2)| \leq C_2 d(f_1, f_2)$.*

As a result of this, for any $\pi \in \Delta_m$ and $f \in \mathcal{F}$, we have $C_1 d(f, f_\pi) \leq L(\pi, f) - L(\pi, f_\pi) \leq C_2 d(f, f_\pi)$. Therefore, if it is known that $L(\pi, f) - L(\pi, f_\pi) \leq \epsilon$, then we can conclude that we must have $d(f, f_\pi) \leq \epsilon/C_1$. Hence, by the strong-density *assumption on $P_{X|Z=z}$ we have that $\pi(z) \times |L(z, f) - L(z, f_\pi)| \leq 1 \times C_2 d(f, f_\pi) \leq (C_2/C_1)\epsilon := 2C\epsilon$.*

Our final assumption is a standard uniform convergence requirement which is satisfied by many commonly used families of classifiers (several examples are detailed in Remark 1).

Assumption 4. *Let $\delta \in [0,1]$ be a confidence parameter. For each $z \in \mathcal{Z}$, there exists a monotonically non-increasing sequence, $\{e_z(N, \mathcal{F}, \delta) : N \geq 1\}$, with $\lim_{N\to\infty} e_z(N, \mathcal{F}, \delta) = 0$, such that the following event holds with probability at least $1 - \delta/2$:*

$$\Omega = \bigcap_{z \in \mathcal{Z}} \bigcap_{N=1}^{\infty} \Big\{ \sup_{f \in \mathcal{F}} \big|\widehat{L}_N(z,f) - L(z,f)\big| \leq e_z(N, \mathcal{F}, \delta) \Big\},$$

where $\widehat{L}_N(z,f) := \frac{1}{N}\sum_{i=1}^{N} \ell\big(f, X_z^{(i)}, Y_z^{(i)}\big)$, $\forall z \in \mathcal{Z}$, with $\big(X_z^{(i)}, Y_z^{(i)}\big)_{i=1}^{N}$ being an i.i.d. sequence of input-label pairs from P_z.

In what follows, we will drop the $\mathcal{F}$ and δ dependence and refer to $e_z(N, \mathcal{F}, \delta)$ as $e_z(N)$ for all $z \in \mathcal{Z}$ and $N \geq 1$. Given an appropriate sequence of $e_z(N)$, we construct the UCB for the risk function $L(z, f_{\pi_t})$, defined by Equation (2), with the following expression:

$$U_t(z, \hat{f}_t) := \frac{1}{|\mathcal{D}_z|} \sum_{(x,y) \in \mathcal{D}_z} \ell(\hat{f}_t, x, y) \; + \; e_z\left(N_{z,t}\right) + \frac{2C}{\pi_t(z)} \sum_{z' \in \mathcal{Z}} \pi_t(z') e_{z'}(N_{z',t}) \; . \tag{4}$$

Where C is the constant described in Assumption 3. The first term on the RHS is the empirical loss of the learned classifier at round t on attribute $Z = z$, so by Assumption 4 the first two terms of the RHS of Equation 4 give a high probability upper bound on $L(z, \hat{f}_t)$, the expected loss of $\hat{f}_t$ conditioned on $Z = z$. The third term then provides an upper bound on the difference $|L(z, f_{\pi_t}) - L(z, \hat{f}_t)|$ and so altogether this provides the desired UCB on the expected loss of the π_t optimal classifier on attribute $Z = z$. The form of the third term is due to Assumption 3 and is discussed in detail in Appendix B.1.

3.1 Theoretical Analysis

In this section, we derive an upper-bound on the excess risk $\mathcal{R}_n(\mathcal{A}_{\texttt{opt}})$ of Algorithm 2. We also show that the performance achieved this algorithm cannot in general be improved, by obtaining an algorithm independent lower-bound on the excess risk for a particular class of problems.

Upper Bound. We begin by obtaining an upper bound on the convergence rate of the excess risk of $\mathcal{A}_{\texttt{opt}}$, the proof of this result is in Appendix B.1.

Theorem 1. *Let Assumptions 1-3 hold and define $\pi_{\min} := \min_{z \in \mathcal{Z}} \pi^*(z)$. Fix any A such that $\pi_{\min}/2 \leq A < \pi_{\min}$. Suppose the query budget n is sufficiently large, as defined in Equation 19 in Appendix B.1. Then, with probability $1 - \delta$, the excess risk of Algorithm 2 can be upper-bounded as*

$$\mathcal{R}_n(\mathcal{A}_{\mathit{opt}}) = \max_{z \in \mathcal{Z}} \; L(z, f_{\pi_n}) - M^* = \mathcal{O}\Big(\frac{|\mathcal{Z}|C}{\pi_{\min}} \; \max_{z \in \mathcal{Z}} \; e_z(N_A)\Big), \tag{5}$$

where $M^* = L(z, f_{\pi^*})$, $\forall z \in \mathcal{Z}$ (see Proposition 1) and $N_A = n\pi_{\min}^2/(2\pi_{\min} - A)$.

Remark 1. *The uniform deviation bounds, $e_z(N)$, defined in Assumption 4 and the bound on the excess risk of Algorithm 2 (see Equation 5) can be instantiated for several commonly used classifiers (function classes $\mathcal{F}$) to obtain an explicit convergence rate in n. If $\mathcal{F}$ has a finite VC-dimension, d_{VC}, then a suitable deviation bound is $e_z(N) = 2\sqrt{\left(2d_{VC}\log\left({}^{2eN}/_{d_{VC}}\right) + 2\log\left({}^{2N^2\pi^2|\mathcal{Z}|}/_{3\delta}\right)\right)/N}$, where here π is the numerical constant. And if $\mathcal{F}$ has Rademacher complexity $\mathfrak{R}_n$, we can choose $e_z(N) = 2\mathfrak{R}_N + \sqrt{\log\left({}^{N^2\pi^2|\mathcal{Z}}/_{3\delta}\right)/N}$. Furthermore, with these uniform deviation bounds and if $\mathfrak{R}_n$ is $\mathcal{O}\left(n^{-\alpha}\right)$, for some $\alpha > 0$, then we obtain excess risk bounds of*

$$\mathcal{R}_n(\mathcal{A}_{\mathit{opt}}) = \mathcal{O}\Big(\frac{|\mathcal{Z}|C}{\pi_{\min}^{1.5}} \sqrt{\frac{d_{VC}}{n}}\Big), \qquad \mathcal{R}_n(\mathcal{A}_{\mathit{opt}}) = \mathcal{O}\Big(\frac{|\mathcal{Z}|C}{\pi_{\min}^{1+\alpha} n^{\alpha}}\Big),$$

for the VC-dimension and Rademacher cases, respectively. These conditions on $\mathcal{F}$ are satisfied by several commonly used classifiers, such as linear, SVM, and multi-layer perceptron.

Remark 2 (Analysis of ϵ-`greedy`). *Abernethy et al. (2020) only analyze the greedy version (i.e., $\epsilon = 0$) of their ϵ-`greedy` sampling strategy with $|\mathcal{Z}| = 2$, and show that at time n, either the excess risk is of $\mathcal{O}\Big(\max_{z \in \mathcal{Z}} \sqrt{{}^{2d_{VC}(\ell \circ \mathcal{F})\log(2/\delta)}/_{N_{z,n}}}\Big)$, or the algorithm draws a sample from the attribute with the largest loss. However, due to the greedy nature of the algorithm they analyzed, there are no guarantees that $N_{z,n} = \Omega(n)$, and thus, in the worst case the above excess risk bound is $\mathcal{O}(1)$. We show in Appendix C how the techniques we developed for the analysis of Algorithm 2 can be suitably employed to study their ϵ-`greedy` strategy. In particular, we obtain sufficient conditions on ϵ under which the excess risk of ϵ-`greedy` converges to zero, and the rate at which this convergence occurs.*

Remark 3 (Comparison between $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy`). *$\mathcal{A}_{\mathit{opt}}$ holds two primary advantages over ϵ-`greedy`, these are discussed in detail in Appendix C, but briefly $\mathcal{A}_{\mathit{opt}}$ has superior risk guarantees, by a constant factor. And both algorithms depend on tunable parameters: C for $\mathcal{A}_{\mathit{opt}}$ and ϵ for ϵ-`greedy`, but $\mathcal{A}_{\mathit{opt}}$ is far more robust to the choice of parameter. Figure 9 in Appendix C illustrates a case where poor choice of ϵ can prevent ϵ-`greedy` from converging to the optimal mixture distribution for any number of samples. This robustness makes $\mathcal{A}_{\mathit{opt}}$ easier to employ in practical settings.*

Lower Bound. Let $\mathcal{Q} = (\boldsymbol{\mu}, \mathcal{F}, \ell_{01})$ denote the class of problems where $\boldsymbol{\mu} \in \mathcal{M}$ is an instance of the `SyntheticModelI` described in Section 2.2, $\mathcal{F}$ is the class of linear classifiers in two dimensions, and ℓ_{01} is the $0-1$ loss. For this function class $\mathcal{F}$, $e_z(N) = \mathcal{O}(\sqrt{\log(N)/N})$ for $z \in \mathcal{Z} = \{u, v\}$, which implies that the excess risk achieved by both $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy` strategies is of $\mathcal{O}\left(\sqrt{\log(n)/n}\right)$. We prove in Appendix D that this convergence rate (in terms of n) cannot in general be improved by showing that $\max_{Q \in \mathcal{Q}} \; \mathbb{E}_Q \left[\mathcal{R}_n\left(\mathcal{A}\right)\right] = \Omega\left(1/\sqrt{n}\right)$.

4 Heuristic Extensions

We show in Appendix B.1.1 that if the uniform deviation bounds, $e_z(N)$, can be chosen to decrease to zero sufficiently quickly as N increases, then we can omit the C dependent third term in Equation (4) and still attain the same regret bounds given in Theorem 1. The resulting two-term UCB is then only the high probability upper bound on $L(z, \hat{f}_t)$. We will use this UCB as the basis for several practical modifications to our optimistic adaptive sampling strategy. First we note that in UCB based algorithms the confidence bounds necessary to attain theoretical results often do not produce optimal empirical results–see, for example, Section 6 in Srinivas et al. (2009). So, following standard practice, we introduce a hyperparameter, c_0, in Equation (6) below, which we can tune to optimize the exploration/exploitation trade-off.

Practical applications of our strategy run into two further challenges. As described in Section 3, $\mathcal{A}_{\texttt{opt}}$ requires re-training the classifier in every iteration. While this can be implemented in small problems, it becomes infeasible for problems involving large models, such as CNNs. Also, as noted in Section 2.1, for data where one attribute is significantly easier than the other, Assumption 2 may not hold. In this case it may not be beneficial to continue to sample from the attribute with the largest loss. To address these issues we present a heuristic variant of Algorithm 2.

For the first challenge we make the following modifications to Algorithm 2. **1)** We expand Phase 1 to n_0 rounds, **2)** at each subsequent round we draw two batches of size b_0 from the chosen attribute, **3)** instead of re-training from scratch each round, we update the previous model, $\hat{f}_{t-1}$, and **4)** instead of training to convergence, we perform one gradient step over the entire accumulated training set, $\mathcal{D}_t$.

To address the second challenge we modify the UCB by adding a term based on the Mann-Kendall statistic (Mann, 1945) (Kendall, 1948) which, for time series data $(X_i)_{i=1}^n$, is given by $S = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \operatorname{sgn}\left(X_j - X_i\right)$. This is designed to identify monotonic upwards or downward trends in time series. We calculate the statistic for each attribute, and denote it S_{z_t}, for the accuracy of the classifier, $\hat{f}_t$, on the attribute's validation set, $\mathcal{D}_{z_t}$, over time. Intuitively, this is tracking whether training on additional samples from an attribute is improving the classifier's accuracy on that attribute. We incorporate the statistic into the UCB as follows:

$$\widetilde{U}_t(z, \hat{f}_t) \coloneqq \frac{1}{|\mathcal{D}_z|} \sum_{(x,y) \in \mathcal{D}_z} \ell(\hat{f}_t, x, y) \; + \; \frac{c_0}{\sqrt{N_{z,t}}} \; + c_1 \frac{S_{z_t}}{\sqrt{\mathrm{var(S_{z_t})}}}, \tag{6}$$

to incentivize the algorithm to sample attributes on which accuracy is increasing. c_1 is a free parameter controlling the importance of per-attribute loss trends. We note that the general ability to modify the UCB in this manner is a strength of our algorithm, as it allows for a great deal of interpretability and adaptability in practical applications.
The parameters c_0, c_1, n_0, and b_0 are chosen based on the specific problem and available computational resources. We describe our selection of these terms Appendix E. In the experiments we will refer to the variant of this algorithm with $c_1 = 0$ as $\mathcal{A}_{\texttt{opt}}$, and the variant with $c_1 > 0$ as $\mathcal{A}_{\texttt{opt}}+$.

5 Empirical Results

We evaluate the performance of our proposed active sampling algorithm $\mathcal{A}_{\texttt{opt}}$ on both synthetic and real datasets, and compare it with the following baselines: **1)** ϵ-`greedy` scheme of Abernethy et al. (2020), **2)** `Greedy` scheme, which is equivalent to ϵ-`greedy` with $\epsilon = 0$, **3)** `Uniform`, where an equal number of samples are drawn for each attribute, and **4)** `Uncurated`, where samples are drawn according to the natural distribution of the dataset. We note that, for some datasets, the natural distribution is uniform, for those we omit the results for the `Uncurated` scheme.

We will make the code for these experiments available in the supplemental materials. All experiments were run for multiple trials–the number of which is indicated in the results–and we report the average over trials with shaded regions on plots indicating ±1 standard deviation. Experiments for image datasets were run on a single GPU provided by Google Colab or other shared computing resources.

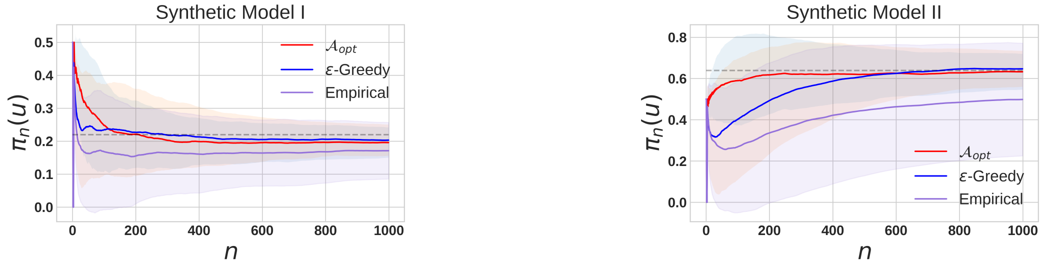

Figure 2: The figure shows the convergence of $\pi_n(u)$ for the three algorithms $\mathcal{A}_{\texttt{opt}}$, ϵ-`greedy` (with $\epsilon = 0.1$) and `Empirical` (i.e., ϵ-`greedy` with $\epsilon = 0$), to the optimal value $\pi^*(u)$ for the two instances of the `SyntheticModel1` introduced in Section 2.2.

Synthetic Dataset. In this experiment, we compare how the $\pi_n(u)$ returned by the different algorithms converge to $\pi^*(u)$ for the two synthetic models introduced in Section 2.2 with $\mathcal{F}$ chosen as the family of logistic regression (LR) classifiers. Since the feature space is two dimensional, we use the version of $\mathcal{A}_{\texttt{opt}}$, ϵ-`greedy`, and `Empirical` schemes in which we train (from scratch) the classifier in each round. For $\mathcal{A}_{\texttt{opt}}$, we use UCB given in 6 with $c_0 = 0.1$ and for ϵ-`greedy`, we use $\epsilon = 0.1$. These values of c_0 and ϵ are selected via a grid search.

Figure 2 shows how $\pi_n(u)$ changes with n for the three algorithms averaged over 100 trials. As expected, the algorithms with an exploration component (i.e., $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy`) eventually converge to the optimal $\pi^*(u)$ value in both cases, whereas the `Empirical` scheme that acts greedily often gets stuck with a wrong mixture distribution, resulting in high variability in its performance.

***Adult* Dataset.** For the remaining experiments we find that, for properly tuned values of ϵ and c_0, both $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy` attain comparable minimum test error. So we omit the ϵ-`greedy` results for the purpose of clarity, see Appendix C for a detailed comparison of the two algorithms.

We now analyze the performance of the remaining algorithms on a dataset from the UCI ML Repository (Dua and Graff, 2017) that is commonly used in the fairness literature: the *Adult* dataset. It is a low dimensional problem, so we use LR classifiers and the exact version of each sampling algorithm, where the optimal classifier is computed with each new sample.

We set $\mathcal{Z}$ to be white men, non-white men, white women, and non-white women. The minimum test accuracy over all attributes at each iteration is displayed in Figure 3a. Each sampling scheme approaches 80.5% accuracy as sample size grows, this matches the results achieved under the LR algorithms reported in Table 4a) in Martinez et al. (2020). This is the maximum accuracy an LR classifier can achieve on the white male sub-population in isolation and so additional sampling, or other fairness algorithms, cannot improve on this performance in a minimax sense. We do note, however, that the adaptive algorithms hold a sizable advantage over `Uniform` at small sample sizes.

Image Datasets. We also compare the performance of the different sampling schemes on larger scale problems and more complex hypothesis classes. To this end we use three image datasets: *UTKFace* (Zhang et al., 2017), *FashionMNIST* (Xiao et al., 2017), and *Cifar10* (Krizhevsky and Hinton, 2009) with CNNs. We use the heuristic variants of the $\mathcal{A}_{\texttt{opt}}$, with $c_1 = 0$, and `Greedy` algorithms with batch sizes of 50. The CNN architecture, data transforms, and further algorithm parameters are detailed in Appendix E.

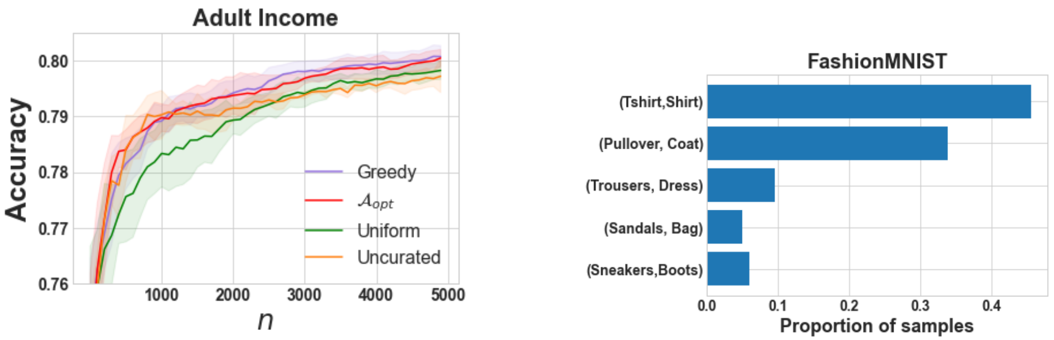

Figure 3: Left: Minimum test accuracy over all attributes for *Adult* dataset as a function of the sampling budget n, averaged over 10 trials for *Adult*. Right: Mixture distribution learned by $\mathcal{A}_{\texttt{opt}}$ over 500 training rounds on *FashionMNIST*.

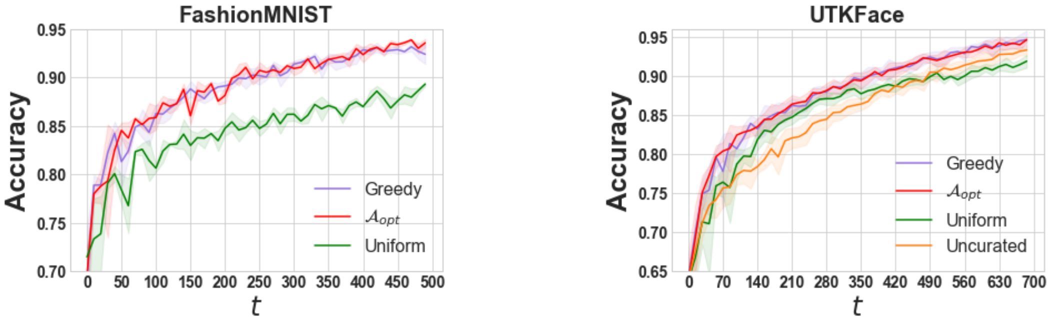

Figure 4: Minimum test accuracy, over all attributes, for both *FashionMNIST* and *UTKFace* as a function of the time step t, averaged over 10 trials.

The *UTKFace* dataset consists of face images annotated with age, gender, and ethnicity. We choose $Y = \{\texttt{Male}, \texttt{Female}\}$ and set $\mathcal{Z}$ to the five ethnicities. The minimum test accuracy, over all attributes, is shown in Figure 4b and demonstrates a clear separation between the adaptive algorithms and both `Uniform` and `Uncurated` sampling. `Uncurated` does particularly poorly in the low sample regime here, in contrast with the *Adult* dataset, where `Uncurated` performed comparably to the adaptive algorithms. This is because, for *Adult* dataset, the lowest accuracy attribute is over-represented in the dataset with white men at 63% of all samples. Whereas, for *UTKFace*, the accuracy was lowest for Asian people, who are under-represented in the available data at only 14% of all samples.

The other two datasets, *FashionMNIST* and *Cifar10*, were chosen to provide a controlled setting to demonstrate the existence of hard and easy attributes on real-world data. For both, we divide the labels into 5 pairs and assign each an "attribute". The pairs were chosen according to existing confusion matrices to have pairs that are both easy and hard to distinguish from each other, see Appendix E for more details.

For *FashionMNIST*, each pair in Figure 3b is one attribute and within each pair one item is assigned $Y = 0$ and the other $Y = 1$. Then a single binary CNN classifier is learned simultaneously over all 5 pairs. Figure 3b shows the mixture distribution generated by the $\mathcal{A}_{\texttt{opt}}$ sampling scheme, which allocated the vast majority of samples to the (Tshirt, Shirt) and (Pullover, Coat) pairs. This makes intuitive sense, since both pairs of items are qualitatively very similar to each other, and aligns with common confusion matrices which indicate that those items are frequently misclassified as each other by standard classifiers. Figure 4a displays the worst case accuracy for each scheme on *FashionMNIST* as a function of time step and shows that both adaptive algorithms outperform `Uniform` sampling throughout the training process. Finally, Figure 5 shows the test accuracy on each attribute for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` sampling schemes. $\mathcal{A}_{\texttt{opt}}$ maintains a much smaller spread between the accuracy over different attributes. This equity is particularly desirable from a fairness perspective as it ensures no attribute has a large advantage over any other in the sense of expected accuracy. It also validates our assumption that the optimal distribution will tend to equalize the losses across attributes in real world data.

Final accuracy for all experiments, per-attribute accuracy for *UTKFace*, all results for *CIFAR10*, and details of the *Adult* dataset are included in Appendix E, along with results for the *German* dataset, another UCI ML Repository dataset.

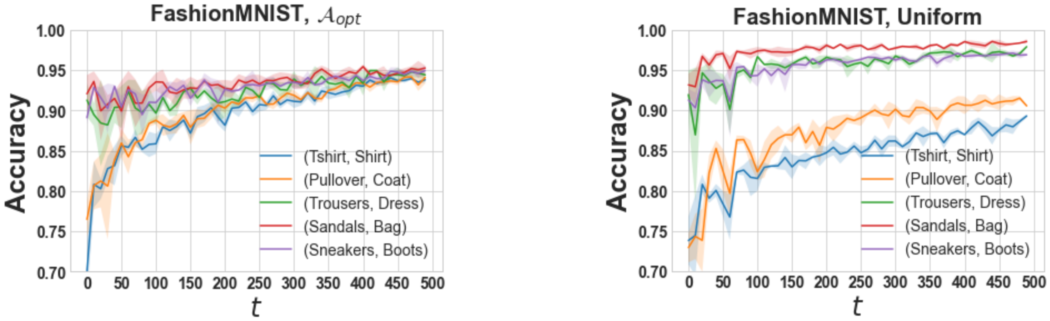

Figure 5: Test accuracy for each attribute in *FashionMNIST* as a function of the time step, t, for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` sampling schemes, averaged over 10 trials.

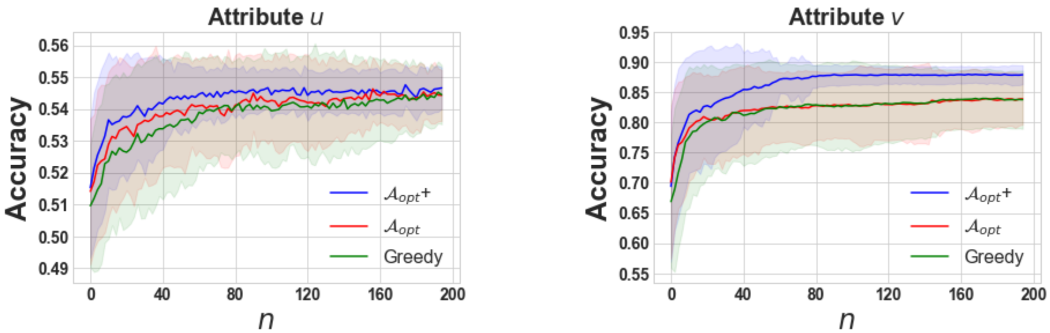

Figure 6: Test accuracy as a function of sampling budget for both attributes from the dataset shown in Fig. 7, averaged over 100 trials.

Empirical results when Assumption 2 is violated.

Finally, we consider the case where our assumption of a unique, risk equalizing mixture distribution may not hold. The practical effects of this situation can be seen in the *Adult* dataset where additional samples of the white male attribute show diminishing returns, while other attributes can attain higher accuracy given more samples. In such a scenario, each adaptive algorithm will continue to select samples mainly from the worst case group despite this granting little improvement in performance.

To evaluate this scenario we introduce a second synthetic data model: an instance of SyntheticModel2 is illustrated in Figure 7 and specified in detail in Definition 4 in Appendix E. In this model we create one attribute, shown in red, which allows a relatively small maximal accuracy that can be attained with few samples. In contrast, the blue attribute can be classified with high accuracy given few sample, as the bulk of its mass is in the central, separable clusters shown in the figure. But classification of this attribute also benefits from many additional samples as the sparser regions in the top left and bottom right are explored. In this setting, constantly sampling from the attribute with the lowest empirical accuracy is inadvisable as additional samples of the red attribute cannot increase its accuracy, while additional samples of the blue attribute can increase its accuracy.

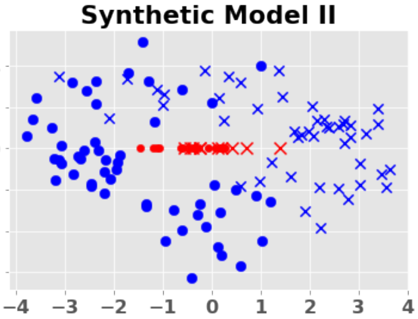

Figure 7: An instance of SyntheticModel2with attribute $Z = u$ shown in red and $Z = v$ shown in blue.

We compared the performance of $\mathcal{A}_{\mathtt{opt}}+$, the heuristic variation of our algorithm with a trend-based statistic included, $\mathcal{A}_{\mathtt{opt}}$, and Greedy on SyntheticModel2. Figure 6 shows the results, each of the three algorithms achieves similar minimax accuracy and performs worst on attribute $Z = u$, shown in red in Figure 7. But $\mathcal{A}_{\mathtt{opt}}+$ achieves 4 points higher average accuracy than both Greedy and $\mathcal{A}_{\mathtt{opt}}$ with the original UCB on the other attribute, $Z = v$. This demonstrates that the additional term in the $\mathcal{A}_{\mathtt{opt}}+$ UCB is effective at recognizing when the hardest group is not benefiting from additional training samples and redistributing them more effectively.

6 Conclusion

We considered the problem of actively constructing a training set in order to learn a classifier that achieves minimax fairness in terms of predictive loss. We proposed a new strategy for this problem ($\mathcal{A}_{\mathtt{opt}}$) and obtained theoretical guarantees on its performance. We then showed that the theoretical performance achieved by $\mathcal{A}_{\mathtt{opt}}$ and ϵ-greedy (Abernethy et al., 2020) cannot be improved in general, by obtaining algorithm independent lower-bounds for the problem.

Our experiments demonstrated that adaptive sampling schemes can achieve superior minimax risk [Fig. 3a, Fig. 4] and smaller disparity between per-attribute risks [Fig. 5] compared to both uniform and uncurated schemes. The results in Fig. 2 show the necessity of exploration, as purely greedy algorithms can converge to sub-optimal mixture distributions. And finally Fig. 6 shows the versatility of our general UCB-based strategy in its ability to readily incorporate new terms to accommodate the practical challenges posed by real-world problems.

Our theoretical results rely on the existence of a unique risk equalizing mixture distribution π^*, a condition that may not always hold. Thus, an important future work is to relax this assumption, and to design and analyze algorithms that identify pareto optimal mixture distributions that achieve minimax fairness. Another important direction is to derive and analyze active sampling strategies for other fairness measures.

7 Acknowledgements

This work was partially supported by ARO under award number W911NF-19-1-0317. We also thank the four anonymous reviewers and the area-chair for their detailed feedback that helped improve the presentation of our results.

References

Abernethy, J., Awasthi, P., Kleindessner, M., Morgenstern, J., and Zhang, J. (2020). Adaptive sampling to reduce disparate performance. *arXiv:2006.06879*.

Agarwal, A., Beygelzimer, A., Dudík, M., Langford, J., and Wallach, H. (2018). A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, pages 60–69.

Anahideh, H., Asudeh, A., and Thirumuruganathan, S. (2020). Fair active learning. *arXiv:2001.01796*.

Audibert, J.-Y. and Tsybakov, A. B. (2007). Fast learning rates for plug-in classifiers. *The Annals of statistics*, 35(2):608–633.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235–256.

Barocas, S. and Selbst, A. (2016). Big data's disparate impact. *California Law Review*, 104.

Caton, S. and Haas, C. (2020). Fairness in machine learning: A survey. *arXiv:2010.04053*.

Celis, L. E., Keswani, V., and Vishnoi, N. (2020). Data preprocessing to mitigate bias: A maximum entropy based approach. In *International Conference on Machine Learning*, pages 1349–1359.

Chen, I., Johansson, F., and Sontag, D. (2018). Why is my classifier discriminatory? In *Advances in Neural Information Processing Systems*.

Diana, E., Gill, W., Kearns, M., Kenthapadi, K., and Roth, A. (2021). Minimax group fairness: Algorithms and experiments. *arXiv:2011.03108v2*.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2012). Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226.

Feldman, M., Friedler, S., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. (2015). Certifying and removing disparate impact. In *proceedings of the 21th international conference on knowledge discovery and data mining*, pages 259–268.

Geoffrion, A. (1968). Proper efficiency and the theory of vector maximization. *Journal of mathematical analysis and applications*, 22(3):618–630.

Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323.

Holstein, K., Vaughan, J. W., III, H. D., Dudík, M., and Wallach, H. M. (2018). Improving fairness in machine learning systems: What do industry practitioners need? *CoRR*, abs/1812.05239.

Hooker, S. (2021). Moving beyond "algorithmic bias is a data problem". *Patterns*, 2(4):100241.

Jo, E. S. and Gebru, T. (2019). Lessons from archives: Strategies for collecting sociocultural data in machine learning. *CoRR*, abs/1912.10389.

Kendall, M. G. (1948). *Rank correlation methods.* Griffin.

Krizhevsky, A. and Hinton, G. (2009). Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario.

Lattimore, T. and Szepesvári, C. (2020). *Bandit algorithms.* Cambridge University Press.

Lipton, Z., Chouldechova, A., and McAuley, J. (2017). Does mitigating ML's impact disparity require treatment disparity? *arXiv:1711.07076*.

Mann, H. B. (1945). Nonparametric tests against trend. *Econometrica*, 13(3):245–259.

Martinez, N., Bertran, M., and Sapiro, G. (2020). Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning*, pages 6755–6764.

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., and Gebru, T. (2018). Model cards for model reporting. *CoRR*, abs/1810.03993.

Polyanskiy, Y. and Wu, Y. (2015). Lecture notes on information theory.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. W. (2009). Gaussian process bandits without regret: An experimental design approach. *CoRR*, abs/0912.3995.

Suresh, H. and Guttag, J. V. (2019). A framework for understanding unintended consequences of machine learning. *CoRR*, abs/1901.10002.

Ustun, B., Liu, Y., and Parkes, D. (2019). Fairness without harm: Decoupled classifiers with preference guarantees. In *Proceedings of the 36th International Conference on Machine Learning*, pages 6373–6382.

Woodworth, B., Gunasekar, S., Ohannessian, M., and Srebro, N. (2017). Learning non-discriminatory predictors. In *Proceedings of the 2017 Conference on Learning Theory*, pages 1920–1953.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. (2013). Learning fair representations. In *International Conference on Machine Learning*, pages 325–333.

Zhang, Song, and Qi (2017). Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

A Proof of Proposition 1

Proof. First we note that there must exist at least one optimum mixture distribution π^* since by the Assumption 1, the objective function is continuous and the domain of optimization, Δ_m, is compact. We also note that any optimal π^* must lie in the interior of the simplex Δ_m as a consequence of Assumption 2 (this follows by contradiction).

Next, we show that any such π^* must equalize the $L(z, f_{\pi^*} := \mathbb{E}_z\left[\ell\left(f_{\pi^*}, X, Y\right)\right]$ for all $z \in \mathcal{Z}$. Indeed, assume that this is not the case and enumerate the elements of $\mathcal{Z}$ as $z_1, \ldots, z_m$ in decreasing order of $L(z, f_{\pi^*})$ value. Also introduce ϱ to denote $\min_{z \neq z_1} L(z_1, f_{\pi^*}) - L(z, f_{\pi^*})$ and (for now) assume that $\varrho > 0$.

Now define π_ϵ as follows: $\pi_\epsilon(z) = \pi^*(z)$ for $z \in \mathcal{Z}' := \mathcal{Z} \setminus \{z_1, z_m\}$, $\pi_\epsilon(z_1) = \pi^*(z_1) + \epsilon$ and $\pi_\epsilon(z_m) = \pi^*(z_m) - \epsilon$. Due to the continuity assumption on the mapping $\pi \mapsto L(z, f_\pi)$ for all $z \in \mathcal{Z}$ in Assumption 1, we note that there exists $\epsilon_0 < \pi_{\min} := \min_{z \in \mathcal{Z}} \pi^*(z)$, such that for all $z \neq z_1$ we have $L(z, f_{\pi_\epsilon}) \leq L(z_1, f_{\pi^*}) - \varrho/2$ for all $\epsilon \leq \epsilon_0$. Note that $\pi_{\min}$ must be strictly greater than 0, since π^* lies in the interior of Δ_m. Finally, due to the monotonicity assumption in Assumption 1, there must exist $\varrho_1 > 0$ such that $L(z_1, f_{\pi_{\epsilon_0}}) = L(z_1, f_{\pi^*}) - \varrho_1 < L(z_1, f_{\pi^*})$. Together, these results imply that

$$\max_{z \in \mathcal{Z}} L(z, f_{\pi_{\epsilon_0}}) \leq \max\{L(z_1, f_{\pi^*}) - \varrho/2,\ L(z_1, f_{\pi^*}) - \varrho_1\} < L(z_1, f_{\pi^*}),$$

thus contradicting the optimality assumption on π^*. This completes the proof of the statement that any optimal π^* must ensure that $L(z, f_{\pi^*}) = L(z', f_{\pi^*})$ for all $z, z' \in \mathcal{Z}$.

Finally, the fact that $L(z, f_{\pi^*}) = L(z', f_{\pi^*})$ for any optimal π^* also ensures the uniqueness of the optimum mixture distribution. Again, assume that this is not the case and there exists another optimal $\tilde{\pi}^* \neq \pi^*$ achieving optimum value in (1). Then there must exist a z such that $\pi(z) \neq \pi^*(z)$, and hence $L(z, f_{\pi^*}) \neq L(z, f_{\tilde{\pi}^*})$. Hence, by Assumption 1, we have $\max_{z \in \mathcal{Z}} L(z, f_{\pi^*}) \neq \max_{z \in \mathcal{Z}} L(z, f_{\tilde{\pi}^*})$, which contradicts our hypothesis that both π^* and $\tilde{\pi}^*$ achieve the optimum value in (2). This concludes the proof of the uniqueness of π^*. □

B Details of Algorithm 2

The detailed pseudo-code of $\mathcal{A}_{\texttt{opt}}$ is given in Algorithm 2.

B.1 Proof of Theorem 1

Recall the UCB first defined in Equation 4:

$$U_t(z, \hat{f}_t) := \frac{1}{|\mathcal{D}_z|} \sum_{(x,y) \in \mathcal{D}_z} \ell(\hat{f}_t, x, y) \;+\; e_z\left(N_{z,t}\right) + \frac{2C}{\pi_t(z)} \underbrace{\sum_{z' \in \mathcal{Z}} \pi_t(z') e_{z'}(N_{z',t})}_{:=\rho_t}. \tag{7}$$

With the parameter C as defined in Assumption 3 and sequences $e_z(N)$ satisfying Assumption 4.

Before proceeding further, we introduce the following notation:

- For any $z \in \mathcal{Z}$ and at any time $t \leq n$, we use $N_{z,t}$ to refer to the number of samples drawn corresponding to the feature z prior to time t, i.e., $N_{z,t} = \sum_{i=1}^{t-1} \mathbb{1}_{\{z_i = z\}}$.
- $\widehat{L}_t(z, f) = \frac{1}{N_{z,t}} \sum_{(x,y) \in \mathcal{D}_{(z)}} \ell(f, x, y)$, denotes the empirical risk of a classifier f on the samples corresponding to attribute z,
- $L(z, f) = \mathbb{E}_z\left[\ell(f, X, Y)\right]$ denotes the population risk of classifier f corresponding to attribute z.
- $\rho_t = \sum_{z \in \mathcal{Z}} \pi_t(z) e_{z, N_{z,t}}$ will be used to quantify the uniform deviation of the classifier $\hat{f}_t$ from the corresponding optimal classifier f_{π_t}. Recall that π_t is the mixture distribution of the $|\mathcal{Z}|$ attributes at time t constructed by the algorithm, i.e., $\pi_t(z) = \frac{N_{z,t}}{t}$ for $z \in \mathcal{Z}$.

Algorithm 2: Optimistic Sampling for Fair Classification ($\mathcal{A}_{\texttt{opt}}$)

```
Input: $n$ (budget), $\mathcal{F}$ (function class), $\ell$ (loss function), $\xi \in (0,1)$ (forced exploration term)
Initialize: $\mathcal{D}_0 = \emptyset$; $e_{z,1} = +\infty$ and $\mathcal{D}_{(z)} = \emptyset$, for all $z \in \mathcal{Z}$;
/* Draw two independent samples from each $z \in \mathcal{Z}$ */
for $t = 1, \ldots, m$ do
    $z \leftarrow z_t \in \mathcal{Z}$;
    $\left(X_t^{(i)}, Y_t^{(i)}\right)_{i=1,2} \sim P_z$;   $\mathcal{D}_{(z)} = \{(X_t^{(1)}, Y_t^{(1)})\}$;   $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(X_t^{(2)}, Y_t^{(2)})\}$;
end
$\pi_t \leftarrow \left(\frac{1}{m}, \ldots, \frac{1}{m}\right)$;   $\hat{f}_t \in \arg\min_{f \in \mathcal{F}} \frac{1}{t} \sum_{(x,y) \in \mathcal{D}_t} \ell(f, x, y)$;   $N_{z,t} \leftarrow 1, \; \forall z \in \mathcal{Z}$;
for $t = m+1, \ldots, n$ do
    /* choose the next distribution $P_{XY|Z=z}$ to sample */
    if $\min_{z \in \mathcal{Z}} \; N_{z,t} < t^{\xi}$ then
        $z_t \in \arg\min_{z \in \mathcal{Z}} \; N_{z,t}$        // Forced exploration
    else
        $z_t = \arg\max_{z \in \mathcal{Z}} \; U_t(z, \hat{f}_t)$
    end
    /* Collect data */
    $\left(X_t^{(i)}, Y_t^{(i)}\right)_{i=1,2} \sim P_{z_t}$
    /* Perform the updates */
    Update $\left(e_z(N_{z,t})\right)_{z \in \mathcal{Z}}$,
     $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(X_t^{(1)}, Y_t^{(1)})\}$,   $\mathcal{D}_{(z_t)} \leftarrow \mathcal{D}_{(z_t)} \cup \{(X_t^{(2)}, Y_t^{(2)}\}$.
    Update $\pi_t \leftarrow \frac{t-1}{t}\pi_t + \frac{1}{t}\mathbb{1}_{\{z_t\}}$
    /* Update the classifier $\hat{f}_t$ */
    $\hat{f}_t \in \arg\min_{f \in \mathcal{F}} \; \frac{1}{t} \sum_{(x,y) \in \mathcal{D}_t} \; \ell(f, x, y)$
end
Output: $\pi_n$
```

Suppose at the end of n rounds, the resulting mixture distribution is π_n. Then introduce the set of *over-represented* attributes, $\mathcal{Z}_o \subset \mathcal{Z}$, defined as $\mathcal{Z}_o := \{z \in \mathcal{Z} : \pi_n(z) > \pi^*(z)\}$ and refer to the set $\mathcal{Z}_u := \mathcal{Z} \setminus \mathcal{Z}_o$ as the *under-represented* attributes. Note that $\mathcal{Z}_o$ is empty only if $\pi_n = \pi^*$. In this case, the sampling algorithm has learned the optimal mixing distribution resulting in zero excess risk. Hence, for the rest of the proof, we will assume that $\mathcal{Z}_o$ is non-empty. Note that for any $z_0 \in \mathcal{Z}_o$, Assumption 1 ensures that we must have $L(z_0, f_{\pi_n}) < L(z_0, f_{\pi^*})$ since $\pi_n(z_0) > \pi^*(z_0)$.

Lemma 1. *Suppose $t_0 \leq n$ denotes the last time that Algorithm 2 queried any feature belonging to the subset $\mathcal{Z}_o$, and denote the corresponding feature by z_0. Then, if n is large enough to ensure that $n^{1-\xi} \geq \left(\frac{1}{\pi_{\min}}\right)$ where $\pi_{\min} = \min_{z \in \mathcal{Z}} \pi^*(z)$ and that $2\max_{z \in \mathcal{Z}} e_z\left((\pi_{\min} n)^{\xi}\right) \leq \epsilon_0$ (introduced in Assumption 3), we have the following with probability at least $1-\delta$:*

$$L(z, f_{\pi_{t_0}}) \leq \underbrace{L(z, f_{\pi^*})}_{:=M^*} + \underbrace{2e_{z_0}(N_{z_0,t_0}) + \left(4C/\pi^*(z_0)\right)\rho_{t_0}}_{:=B_0}. \tag{8}$$

Recall that $\rho_t = \sum_z \pi_t(z) e_z(N_{z,t})$, C is the constant from Assumption 3, and π^* is the optimal mixture distribution defined in (2)*.*

Proof. Throughout this proof we assume that the $1 - \delta/2$ probability event introduced while defining the UCB term in (4) occurs for the two sequence of samples drawn by Algorithm 2: the first used in updating $\mathcal{D}_t$ for training the classifier $\hat{f}_t$, and the second used in updating $(\mathcal{D}_z)_{z \in \mathcal{Z}}$ for estimating the loss of $\hat{f}_t$. Thus, both of these events occur with probability at least $1 - \delta$.

First note the following chain of inequalities: $\pi_{t_0}(z_0) \geq \pi_n(z_0) > \pi^*(z_0) \geq \pi_{\min}$. Then at time t_0, it must be the case that $N_{z_0,t_0} \geq \pi_{\min} n$ as z_0 belongs to $\mathcal{Z}_o$. This, along with the fact that

$n \geq (1/\pi_{\min})^{1/(1-\xi)}$ means that $N_{z_0,t_0} \geq \pi_{\min} n \geq n^{-1+\xi} n = n^{\xi}$, and hence the query to z_0 must have been made due to the condition in Line 11 of Algorithm 2, and not due to the forced exploration step in Line 9.

Next, we have the following chain of inequalities for any $z \neq z_0$:

$$\begin{aligned} L(z, f_{\pi_{t_0}}) &\overset{\text{(a)}}{\leq} L(z, \hat{f}_{t_0}) + |L(z, \hat{f}_{t_0}) - L(z, f_{\pi_{t_0}})| \\ &\overset{\text{(b)}}{\leq} L(z, \hat{f}_{t_0}) + \left({}^{2C}/_{\pi_{t_0(z)}}\right) \rho_{t_0} \\ &\overset{\text{(c)}}{\leq} \widehat{L}_{t_0}(z, \hat{f}_{t_0}) + e_z(N_{z,t}) + \left({}^{2C}/_{\pi_{t_0(z)}}\right) \rho_{t_0} \\ &\overset{\text{(d)}}{\leq} \widehat{L}_{t_0}(z_0, \hat{f}_{t_0}) + e_{z_0}(N_{z_0,t}) + \left({}^{2C}/_{\pi_{t_0}(z_0)}\right) \rho_{t_0} \\ &\overset{\text{(e)}}{\leq} L(z_0, \hat{f}_{t_0}) + 2e_{z_0}(N_{z_0,t_0}) + \left({}^{2C}/_{\pi_{t_0}(z_0)}\right) \rho_{t_0} \\ &\overset{\text{(f)}}{\leq} L(z_0, f_{\pi_{t_0}}) + 2e_{z_0}(N_{z_0,t_0}) + \left({}^{4C}/_{\pi_{t_0}(z_0)}\right) \rho_{t_0} \\ &\overset{\text{(g)}}{\leq} L(z_0, f_{\pi^*}) + 2e_{z_0}(N_{z_0,t_0}) + \left({}^{4C}/_{\pi^*(z_0)}\right) \rho_{t_0} \\ &\overset{\text{(h)}}{=} L(z, f_{\pi^*}) \; + \; 2e_{z_0}(N_{z_0,t_0}) + \left({}^{4C}/_{\pi^*(z_0)}\right) \rho_{t_0} \\ &= M^* + B_0. \end{aligned}$$

In the above display:
(a) follows from an application of triangle inequality,
(b) applies Assumption 3 and uses the fact that n is large enough to ensure that the suboptimality of $\hat{f}_{t_0}$ w.r.t. $f_{\pi_{t_0}}$ is no larger than ϵ_0.
(c) follows from the definition of event Ω_2,
(d) follows from the attribute selection rule in Line 11,
(e) again uses the uniform deviation event Ω_2,
(f) follows from another application of Assumption 3,
(g) uses the fact that $\pi_{t_0}(z_0) > \pi^*(z_0)$, the monotonicity condition from Assumption 1,
(h) uses the result of Proposition 1 to write $L(z_0, f_{\pi^*}) = L(z, f_{\pi^*}) := M^*$. □

Next, we show that for a large enough value of n, due to the forced exploration step (Line 9 in Algorithm 2), the values of $\pi_{t_0}(z)$ for $z \in \mathcal{Z}_u$ are not too small.

Lemma 2. *For any given $A < \pi_{\min} = \min_{z \in \mathcal{Z}} \pi^*(z)$, there exists an $n_0 < \infty$, defined in* (12) *below, such that for all $n \geq n_0$, the following are true at time t_0 (recall that t_0 denotes the last time at which an attribute from $\mathcal{Z}_o$ was queried by the algorithm):*

$$\pi_{t_0}(z) \geq \frac{(m-1)\pi^*(z)}{m} + \frac{A}{m}, \qquad \textit{for all } \; z \in \mathcal{Z}_u, \tag{9}$$

$$\textit{and} \quad t_0 \geq n \left(\frac{\pi_{\min}}{2\pi_{\min} - A} \right). \tag{10}$$

Recall that in the above display $m = |\mathcal{Z}|$.

Proof. First note that for any $z \in \mathcal{Z}_u$, we have

$$M^* \leq L(z, f_{\pi_{t_0}}) \leq M^* + B_0,$$

where we have used the notation $M^* = L(z, f_{\pi^*})$ introduced in (8). The left inequality above is due to the monontonicity condition of Assumption 1, while the right inequality is from Lemma 1.

Introducing the notation $\pi_{\min} = \min_{z \in \mathcal{Z}} \pi^*(z)$, we note that by definition $t_0 \geq \pi_{\min} n$. Recall that the term B_0 is defined as $B_0 = 2e_{z_0}(N_{z_0,t_0}) + \left({}^{4C}/_{\pi^*(z_0)}\right) \rho_{t_0}$. Now, since $N_{z_0,t_0} \geq \pi_{\min} n$ and due to the monotonicity of $e_z(N_{z,t})$, we can upper bound $e_{z_0}(N_{z_0,t_0})$ with $e_{z_0}(\pi_{\min} n)$. Next, recall that $\rho_{t_0} = \sum_{z \in \mathcal{Z}} \pi_{t_0}(z) e_z(N_{z,t_0}) \leq \max_{z \in \mathcal{Z}} e_z(N_{z,t_0})$. Since $t_0 \geq N_{z_0,t_0} \geq \pi_{\min} n$, and due to the fact that forced-exploration step, i.e., Line 9 of Algorithm 2, was not needed at time t_0, we must have $N_{z,t_0} \geq (\pi_{\min} n)^{\xi}$ for all $z \in \mathcal{Z}_u$. Thus the second term in the definition of B_0 can be simply bounded with ${}^{4C}/_{\pi^*(z_0)} \max_{z\in\mathcal{Z}} e_z((\pi_{\min}n)^{\xi}) \leq {}^{4C}/_{\pi_{\min}} \max_{z\in\mathcal{Z}} e_z((\pi_{\min}n)^{\xi})$. Combining these two steps, we finally get that $B_0 \leq ({}^{4C}/_{\pi_{\min}} + 2) \max_{z\in\mathcal{Z}} e_z((\pi_{\min}n)^{\xi})$.

By the monotonicity of the terms $e_z(N_{z,t})$ and ρ_t, we note that $\lim_{n\to\infty} B_0 = 0$, since $\lim_{n\to\infty} \max_{z\in\mathcal{Z}} e_z((\pi_{\min}n)^{\xi}) = 0$. Thus as n goes to infinity, $L(z, f_{\pi_{t_0}})$ converges to the optimal value M^*, which by continuity of the mapping $\pi \mapsto L(z, f_\pi)$ for all $z \in \mathcal{Z}$ implies that $\pi_{t_0} \to \pi^*$. We can use this fact to define a sufficient number of samples, denoted by n_0, beyond which it can be ensured that π_{t_0} satisfies the statement in (9).

$$\pi_{\min} = \min_{z\in\mathcal{Z}} \pi^*(z); \qquad b = \inf\left\{\max_{z\in\mathcal{Z}} L(z, f_\pi) - M^* : \|\pi^* - \pi\|_\infty > \frac{\pi_{\min} - A}{m}\right\}; \tag{11}$$

$$n_0 \coloneqq \max\left\{n_0',\ \frac{1}{\pi_{\min}^2}\right\}; \qquad n_0' \coloneqq \min\left\{n \geq 1 : \left(\frac{4C}{\pi_{\min}} + 2\right) \max_{z\in\mathcal{Z}} e_z((\pi_{\min}n)^{\xi}) \ \leq\ b\right\}. \tag{12}$$

Thus the definition of the term b, combined with the upper bound on B_0 due to Lemma 1 and the forced-exploration rule ensure that for $n \geq n_0$, we must have $\pi_{t_0}(z) \geq \pi^*(z)/2$ for all $z \in \mathcal{Z}_u$ as required by (9).

Since we will use the above computation of n_0' several times, we formalize it in terms of the following definition.

Definition 2 (`SmallestBudget`)**.** Given constants $c > 0, p, q$ and $r \in (0, 1]$, the function `SmallestBudget` returns the following

$$\texttt{SmallestBudget}(c, p, q, r) = \min\left\{n \geq 1 : \max_z e_z(N_{pq}) \ \leq\ \gamma/c\right\}, \quad \text{where}$$

$$\gamma = \inf\left\{\max_{z\in\mathcal{Z}} L(z, f_\pi) - M^* : \|\pi^* - \pi\|_\infty > r\right\} \quad \text{and} \quad N_{pq} = (pn)^q.$$

Note that n_0' in (12) is equal to `SmallestBudget` $((4C/\pi_{\min} + 2),\ \pi_{\min},\ \xi,\ (\pi_{\min} - A)/m)$.

For the proof of the statement in (10), we note that since $\pi_{t_0}(z) \geq \pi^*(z) - \frac{\pi_{\min} - A}{m}$ for all $z \neq z_0$, the time t_0 must satisfy the following:

$$\begin{aligned} t_0 = N_{z_0,t_0} + \sum_{z\neq z_0} N_{z,t_0} &\geq \pi^*(z_0)n + \sum_{z\neq z_0}\left(\pi^*(z) - \frac{\pi_{\min} - A}{m}\right) t_0 \\ &= \pi^*(z_0)n + \left(1 - \pi^*(z_0) + \frac{(m-1)(\pi_{\min} - A)}{m}\right) t_0. \end{aligned}$$

This, in turn, implies that

$$t_0 \geq n\left(\frac{\pi^*(z_0)}{\pi^*(z_0) + \frac{m-1}{m}(\pi_{\min} - A)}\right) \geq n\left(\frac{\pi_{\min}}{2\pi_{\min} - A}\right).$$

This completes the proof of (10). □

We now state a basic result about the behavior of π_t:

Lemma 3. *Suppose the empirical mixture distribution is π_r at some time r, and in the time interval $\{r+1, \ldots, t\}$ the agent only queries attributes z from $\mathcal{Z}' \subset \mathcal{Z}$ such that $\sum_{z\in\mathcal{Z}'} \pi_r(z) < 1$. Then there exists at least one $z \in \mathcal{Z}'$ such that $\pi_t(z) > \pi_r(z)$.*

Proof. The statement follows by contradiction. Assume that the conclusion stated above is not true, and $\pi_t(z) \leq \pi_r(z)$ for all $z \in \mathcal{Z}'$. Introducing b_z to denote the number of times attribute $z \in \mathcal{Z}'$ is queried in the time interval $\{r+1, \ldots, t\}$, we then have

$$\pi_t(z) = \frac{r\pi_r(z) + b_z}{t} \ \Rightarrow\ (t - r)\pi_t(z) + r\underbrace{(\pi_t(z) - \pi_r(z))}_{\leq 0} = b_z \ \Rightarrow\ (t - r)\pi_t(z) \geq b_z. \tag{13}$$

Note that $\sum_{z\in\mathcal{Z}'}\pi_t(z) = \frac{t-r(1-\sum_z \pi_r(z))}{t} < 1$ by assumption that $\sum_{z\in\mathcal{Z}'}\pi_r(z) < 1$. Combining this with (13), we get the required contradiction as follows:

$$(t-r) > (t-r)\sum_{z\in\mathcal{Z}'}\pi_t(z) \geq \sum_{z\in\mathcal{Z}'} b_z = (t-r).$$

□

Before proceeding, we introduce some notations. As stated earlier, due to the definition of the term t_0, we know that in the rounds $t \in \{t_0+1,\ldots,n\}$, the algorithm only queries the attributes belonging to the set $\mathcal{Z}_u$. If the set $\mathcal{Z}_u$ is empty, that means t_0 must be equal to n and the algorithm stops there. Otherwise, the interval $\{t_0+1,\ldots,n\}$ can be partitioned into $\{t_0+1,\ldots,t_1\}$, $\{t_1+1,\ldots,t_2\}$, $\ldots$, $\{t_s+1,\ldots,n\}$ for appropriately defined $t_1,\ldots,t_s$ and $s \leq |\mathcal{Z}|-1$ as follows.

- First we introduce the term $\mathcal{Z}_t$ to denote the 'active set' of attributes at time t, i.e., the set of attributes that are queried at least once after time t. Note that we have $\mathcal{Z}_{t_0} = \mathcal{Z}_u$.
- Then (for $t \geq t_0$) we define a subset $\mathcal{Z}_o^{(1)}$ of $\mathcal{Z}_t$ as those attributes $z \in \mathcal{Z}_t$ for which we have $\pi_{t_0}(z) < \pi_n(z)$. By Lemma 3, we know that $\mathcal{Z}_o^{(1)}$ must be non-empty.
- Next, we define t_1 as the last time $t \leq n$ at which an attribute $z \in \mathcal{Z}_o^{(1)}$ is queried by the algorithm.
- If $t_1 = n$, then we stop and $s = 1$. Otherwise, we repeat the previous two steps with $\mathcal{Z}_{t_1} = \mathcal{Z}_{t_0} \setminus \mathcal{Z}_o^{(1)}$.

To clarify the above introduced notation, we present an example.

Example 2. *Consider a problem with set of attributes $\mathcal{Z} = \{\mathfrak{a}_1,\mathfrak{a}_2,\mathfrak{a}_3,\mathfrak{a}_4\}$, $n = 20$ and $\pi^* = (0.25, 0.25, 0.25, 0.25)$. Suppose an adaptive algorithm*[3] *selects the following sequence of attributes (given a budget of* 20*):*

$$\mathfrak{a}_1,\ \mathfrak{a}_2,\ \mathfrak{a}_3,\ \mathfrak{a}_4,\ \mathfrak{a}_1,\ \mathfrak{a}_1,\ \mathfrak{a}_1,\ \mathfrak{a}_1,\ \mathfrak{a}_1,\ \mathfrak{a}_1,\ \mathfrak{a}_4,\ \mathfrak{a}_1,\ \mathfrak{a}_1,\quad \mathfrak{a}_2,\ \mathfrak{a}_2,\ \mathfrak{a}_2,\ \mathfrak{a}_3,\ \mathfrak{a}_3,\ \mathfrak{a}_3,\quad \mathfrak{a}_4. \tag{14}$$

Then as we can see, the algorithm ends up with $\pi_n = (9/20, 1/5, 1/5, 3/20)$.

- *Comparing π_n with π^*, we observe that the set $\mathcal{Z}_o$ is $\{\mathfrak{a}_1\}$ and $\mathcal{Z}_u = \{\mathfrak{a}_2,\mathfrak{a}_3,\mathfrak{a}_4\}$.*
- *The last time an element of $\mathcal{Z}_o$ is queried, that is t_0, is equal to* 13 *and the corresponding attribute is $z_0 = \mathfrak{a}_1$. The mixture distribution at time t_0 is $\pi_{t_0} = (9/13, 1/13, 1/13, 2/13)$.*
- *Comparing π_{t_0} with π_n, we observe that the set $\mathcal{Z}_o^{(1)}$ is $\{\mathfrak{a}_2,\mathfrak{a}_3\}$ since their fractions increase in the period $[t_0+1,n]_{\mathbb{N}}$, and $\mathcal{Z}_u^{(1)} = \{\mathfrak{a}_4\}$. The last time an element of $\mathcal{Z}_o^{(1)}$ is queried is $t_1 = 19$, and the corresponding attribute is $z_1 = \mathfrak{a}_3$.*
- *Finally, since only one element remains, we have $\mathcal{Z}_o^{(2)} = \{\mathfrak{a}_4\}$, $\mathcal{Z}_u^{(2)} = \emptyset$ and thus $t_2 = n$ and $z_2 = \mathfrak{a}_4$. Also note that the total number of phases above is* 3 *and hence the term $s = 3-1 = 2$.*

Lemma 4. *Suppose, z_1 is the element of $\mathcal{Z}_o^{(1)}$ (introduced above) queried by the algorithm at time t_1. Then, the following is true at time t_1 for all $z \in \mathcal{Z}\setminus\{z_1\}$*

$$L(z, f_{\pi_{t_1}}) \leq M^* + B_0 + B_1, \qquad \textit{where} \qquad B_1 = 2e_{z_1}(N_{z_1,t_1}) + \left({}^{8C}/_{\pi^*(z_1)}\right)\rho_{t_1} \tag{15}$$

Furthermore, repeating this process till the budget is exhausted, we get for any $z \in \mathcal{Z}$ and an $s < m$

$$L(z, f_{\pi_n}) \leq M^* + \sum_{i=0}^{s} B_i, \qquad \textit{where} \qquad B_i = 2e_{z_i, N_{z_i,t_i}} + \left({}^{8C}/_{\pi_{\min}\pi^*(z_i)}\right)\rho_{t_i}, \quad \textit{for}\ \ i > 1. \tag{16}$$

Here s is the (random) number of 'phases' (see Example 2), and is always upper bounded by $m = |\mathcal{Z}|$.

[3] note that the sequence of attributes have been chosen only for illustrating the notation, and do not satisfy the forced exploration condition in Algorithm 2

Proof. To prove (15), we note that at time t_1 for any $z \neq z_1$, we have

$$
\begin{aligned}
L(z, f_{\pi_{t_1}}) &\overset{\text{(a)}}{\leq} L(z, \hat{f}_{t_1}) + \frac{2C}{\pi_{t_1}(z)}\rho_{t_1} \;\leq\; \widehat{L}_{t_1}(z, \hat{f}_{t_1}) + e_{z,N_{z,t_1}} + \frac{2C}{\pi_{t_1}(z)}\rho_{t_1} \\
&\overset{\text{(b)}}{\leq} \widehat{L}_{t_1}(z_1, \hat{f}_{t_1}) + e_{z_1,N_{z_1,t_1}} + \frac{2C}{\pi_{t_1}(z_1)}\rho_{t_1} \\
&\overset{\text{(c)}}{\leq} L(z_1, f_{\pi_{t_1}}) + 2e_{z_1,N_{z_1,t_1}} + \frac{4C}{\pi_{t_1}(z_1)}\rho_{t_1} \\
&\overset{\text{(d)}}{\leq} L(z_1, f_{\pi_{t_0}}) + 2e_{z_1,N_{z_1,t_1}} + \frac{4C}{\pi_{t_1}(z_1)}\rho_{t_1} \\
&\overset{\text{(e)}}{\leq} L(z_1, f_{\pi_{t_0}}) + 2e_{z_1,N_{z_1,t_1}} + \frac{4C}{\frac{(m-1)\pi^*(z_1)}{m} + \frac{A}{m}}\rho_{t_1} \\
&\overset{\text{(f)}}{=} L(z_1, f_{\pi_{t_0}}) + B_1 \;\leq\; M^* + B_0 + B_1.
\end{aligned}
$$

In the above display
(a) uses Assumption 3 and the event Ω_2 introduced while defining the UCB in (4),
(b) follows from the point selection rule in Line 11 of Algorithm 2,
(c) uses Assumption 3,
(d) uses that from the definition of z_1, we must have $\pi_{t_1}(z_1) \geq \pi_{t_0}(z_1)$, and due to monotonicity assumption (Assumption 1), $L(z_1, f_{\pi_{t_0}}) > L(z_1, f_{\pi_{t_1}})$, and
(e) uses the fact that $\pi_{t_1}(z_1) \geq \pi_{t_0}(z_1) \geq \pi^*(z_1)/2$ proved in (9), and
(f) uses Lemma 1 to bound $L(z_1, f_{\pi_{t_0}})$ with $M^* + B_0$ to get (15).

Now, assume that the budget n satisfies $n \geq n_1 := \max\{n_0, n_1'\}$; where

$$
n_1' = \texttt{SmallestBudget}\left(c_0 + c_1,\; \pi_{\min},\; \xi,\; \frac{2(\pi_{\min} - A)}{m}\right), \qquad \text{where} \tag{17}
$$

$$
c_0 = \frac{4C}{A_0} + 2; \quad c_1 = \frac{4C}{A_1} + 2; \quad \text{and} \quad A_i := \frac{i}{m}A + \frac{(m-i)\pi_{\min}}{m}, \text{ for } 0 \leq i \leq m. \tag{18}
$$

Then by the definition of the `SmallestBudget` function (introduced in Definition 2 in Appendix B.1), we must have $\pi_{t_1}(z) \geq \frac{(m-2)\pi^*(z)}{m} + \frac{2A}{m}$ for all $z \in \mathcal{Z}$.

The proof of (16) essentially follows by repeating the above argument a further $s-1$ times. However, there is one minor difference. In proving (15), specifically in Step **(e)**, we used the fact that $\pi_{t_1}(z_1) \geq A_1 := \frac{A}{m} + \frac{(m-1)\pi_{\min}}{m}$. For $i \geq 2$, we can similarly use the fact that $\pi_{t_i}(z_i) \geq A_i := \frac{iA + (m-1)\pi_{\min}}{m}$. The corresponding requirement on n is that, with

$$
\begin{aligned}
& n \geq n_i := \max\{n_0, \ldots, n_{i-1}, n_i'\}, \qquad \text{where} \qquad\qquad (19) \\
& n_i' = \texttt{SmallestBudget}\left(c_0 + \ldots c_i,\; \pi_{\min},\; \xi,\; \frac{(i+1)(\pi_{\min} - A)}{m}\right), \qquad \text{and} \\
& c_i = \frac{4C}{A_i} + 2, \qquad \text{for all} \quad 0 \leq i \leq m.
\end{aligned}
$$

$\square$

Lemma 5. *Finally, we obtain that* $\max_{z\in\mathcal{Z}} L(z, \hat{f}_{\pi_n}) - M^* = \mathcal{O}\left(\frac{|\mathcal{Z}|C}{\pi_{\min}} \max_{z\in\mathcal{Z}} e_z(N_A)\right)$*, where* $N = A\left(\frac{\pi_{\min}}{2\pi_{\min} - A}\right) n$*.*

Proof. First note that for all t_i, $i = 1, 2, \ldots, s$, we have $\rho_t \leq \max_{z\in\mathcal{Z}} e_z(N_{z,t})$. Now, we note the following: for any $i = 0, \ldots s$, we have $\min_{z\in\mathcal{Z}} \pi_{t_i}(z) \geq A_i \geq A$. Also for all $i = 0, \ldots s$, we also have trivially, using (10), $t_i \geq N_A := \pi_{\min}/(2\pi_{\min} - A)n$. Together these two statements imply the following:

- $B_0 \leq c_0 \max_{z\in\mathcal{Z}} e_z(N_{z,t_0}) \leq c_0 \max_{z\in\mathcal{Z}} e_z(N_A) = \left(\frac{4C}{\pi_{\min}} + 2\right) \max_{z\in\mathcal{Z}} e_z(N_A)$.

- Similarly, for $i \geq 1$ we have $B_i \leq c_i \max_{z \in \mathcal{Z}} e_z(N_{z,t_i}) \leq c_i \max_{z \in \mathcal{Z}} e_z(N_A) \leq \left(\frac{4C}{A_i} + 2\right) \max_{z \in \mathcal{Z}} e_z(N_A)$.

Combining the above two points, we get the following

$$\sum_{i=1}^{s} B_i \leq \sum_{i=1}^{s} \left(\frac{4C}{A_{i-1}} + 2 \right) \max_{z \in \mathcal{Z}} e_z(N_A)$$

which, if $A \geq \pi_{\min}/2$, implies

$$\max_{z \in \mathcal{Z}} L(z, f_{\pi_n}) - M^* = \mathcal{O}\left(\frac{|\mathcal{Z}|C}{\pi_{\min}} \max_{z \in \mathcal{Z}} e_z(N_A) \right)$$

as required. Note that the above result holds under the assumption that n is large enough to ensure that the conditions in (19) is satisfied for all $0 \leq i \leq s$. Since $s \leq m-1$, a sufficient condition for this is that the condition in (19) is satisfied for all $0 \leq i \leq m-1$. □

B.1.1 Knowledge of parameter C

In our analysis above, we did not impose any condition on the forced exploration parameter ξ; instead we used knowledge of the parameter C from Assumption 3. We now show that if for some $0 < \xi < 1$, it is known that $\lim_{N \to \infty} \max_{z \in \mathcal{Z}} \frac{e_z(N^{\xi})}{N^{\xi - 1}} = 0$, then we can remove the C dependent term from (4) and obtain the same guarantees for $\mathcal{A}_{\mathtt{opt}}$ as in Theorem 1.

The proof will follow the same outline as in the previous section, and to avoid repetition, we obtain a result analogous to Lemma 1. In particular, with the same definition of t_0 and z_0 as in Lemma 1 and Lemma 2, the following is true at time t_0:

$$\begin{aligned}
L(z, f_{\pi_{t_0}}) &\leq \widehat{L}_{t_0}(z, f_{\pi_{t_0}}) + e_z(N_{z,t_0}) \\
&\leq \underbrace{\widehat{L}_{t_0}(z, \hat{f}_{t_0}) + e_z(N_{z,t_0})}_{=U_{t_0}(z, \hat{f}_{t_0})} + \frac{2C}{\pi_{t_0}(z)} \rho_{t_0} \\
&\overset{(a)}{\leq} U_{t_0}(z_0, \hat{f}_{t_0}) + \frac{2C}{\pi_{t_0}(z)} \rho_{t_0} \\
&\leq L(z_0, f_{\pi_{t_0}}) + 2e_z(N_{z_0,t_0}) + 2C\rho_{t_0} \left(\frac{1}{\pi_{t_0}(z_0)} + \frac{1}{\pi_{t_0}(z)} \right) \\
&\overset{(b)}{\leq} L(z_0, f_{\pi_{t_0}}) + \left(\frac{2C}{\pi_{\min}} + \frac{2C}{t_0^{\xi-1}} \right) \rho_{t_0} \\
&\overset{(c)}{\leq} L(z_0, f_{\pi_{t_0}}) + \underbrace{\left(\frac{2C}{\pi_{\min}} + \frac{2C}{t_0^{\xi-1}} \right) \max_{z \in \mathcal{Z}} e_z(t_0^{\xi})}_{:=B_0'}
\end{aligned}$$

In the above display,
(a) uses the fact that at time t_0, the attribute z_0 was selected by maximizing $U_{t_0}(z, \hat{f}_{t_0})$,
(b) uses the fact that $\pi_{t_0}(z_0) > \pi_{\min}$ and $\pi_{t_0}(z) \geq t_0^{\xi-1}$ for all $z \neq z_0$ since the forced exploration step was not invoked at time t_0, and
(c) uses the fact that $\rho_{t_0} \leq \max_{z \in \mathcal{Z}} e_z(N_{z,t_0}) \leq \max_{z \in \mathcal{Z}} e_z(t_0^{\xi})$ due to the monotonicity of $e_z(N)$ and the fact that $N_{z,t_0} \geq t_0^{\xi}$ for all $z \in \mathcal{Z}$ at time t_0.

Now, to continue with the rest of the proof as in Appendix B.1, we need that the term B_0' converges to zero as n (and hence, t_0) goes to infinity. The first term of B_0', i.e., $2C \max_z e_z(t_0^{\xi})/\pi_{\min}$, converges to zero from the condition that the uniform confidence bound converges to zero as the number of samples, t_0^{ξ}, goes to infinity; while the second term, $\frac{2C}{t_0^{\xi-1}} \max_z e_z(t_0^{\xi})$ converges to zero from the assumption on ξ made at the beginning of this section. Using this fact, we can then proceed as in Lemmas 2, 4, and 5 to get the final result.

B.2 Relation to Active Learning in Bandits

Our formulation of the minimax fair classification problem diverges from the Active learning in bandit (ALB) problem due to the fact that drawing samples from one attribute can reduce the performance of another. To make the discussion concrete, consider an ALB problem with two distributions $Q_1 \sim N(\mu_1, \sigma_1^2)$ and $Q_2 \sim N(\mu_2, \sigma_2^2)$ for $\mu_1, \mu_2 \in \mathbb{R}$ and $\sigma_1^2, \sigma_2^2 \in (0, \infty)$. Let $N_{i,t}$ denote the number of samples allocated by an agent to Q_i for $i = 1, 2$ and $t \geq 1$. Then, we can construct high probability confidence sequences around $\hat{\mu}_{i,t}$ (the empirical counterparts to μ_i at time t) of non-increasing (in t) lengths $e_{i,t}$ such that $|\hat{\mu}_{i,t} - \mu_i| \leq e_{i,t}$. If at time t, the agent decides to draw a sample from arm 1, it would not change the quality of its estimate of Q_2 at time $t + 1$ and beyond.

On the other hand, let us consider Instance 2 of $\texttt{SyntheticModel1}(\boldsymbol{\mu})$ (introduced in Definition 1) with $\mathcal{F}$ being the set of all linear classifiers passing through origin. This is illustrated in Figure 8, where the circles denote the one-square-deviation region around the mean values. The shaded circles correspond to the attribute $Z = v$, and the circles with dashed boundaries correspond to the label $Y = 0$. For example, $P_{X|YZ}(\cdot|Y = 0, Z = u)$ is denoted by the top left circle. Suppose at some time t_1, the current classifier $\hat{f}_{t_1}$ (represented by the solid gray line passing through origin in Figure 8) has low (resp. high) accuracy for the protected attribute $Z = v$ (resp. $Z = u$). To remedy this, the agent may draw more samples from the distribution of attribute $Z = v$ to skew the training dataset distribution towards $Z = v$. This would result in the updated classifier $\hat{f}_{t_2}$ (the dashed gray line in Figure 8) at some time $t_2 > t_1$ to achieve high accuracy for the attribute $Z = v$. But this increased accuracy for $Z = v$ comes at the cost of a reduction in the prediction accuracy for the attribute $Z = u$, as shown in Figure 8.

The above discussion highlights the key distinguishing feature of our problem from the prior work in ALB. Because of this distinction, the existing analyses of the ALB algorithms do not carry over directly to our case. As a result, to quantify the performance of Algorithm 2, we device a new 'multi-phase' approach which is described in detail in Appendix. B.1.

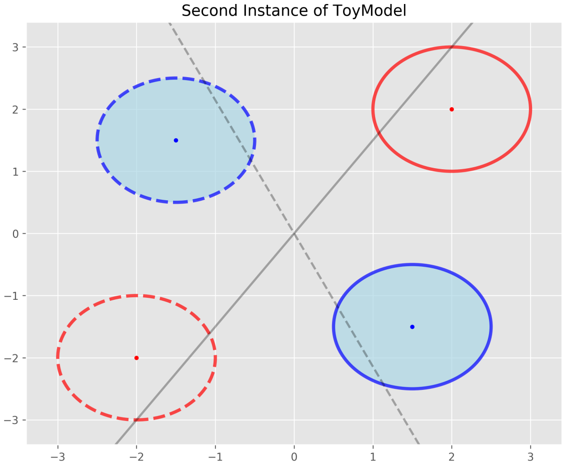

Figure 8: Figure demonstrates the second instance of $\texttt{SyntheticModel1}$ introduced in Section 2.2. Here the circles denote the one-standard-deviation regions of the distributions of features X conditioned on Y and Z. Shaded circles correspond to $Z = u$ and circles with dashed boundaries represent $Y = 0$. The solid gray line represents a possible linear classifier that may be learned if the dataset has fewer examples from $Z = v$. If an adaptive algorithm addresses this by populating the data-set with numerous examples from $Z = v$, a possible updated linear classifier is shown by the dashed gray line. This demonstrates the main distinguishing feature of our problem w.r.t. the active learning in bandits (ALB) problem: *allocating samples to improve the performance on one attribute can have an adverse effect on the performance of the resulting classifier on other attributes*.

C Details of ϵ-`greedy` strategy

The ϵ-`greedy` strategy originally proposed by Abernethy et al. (2020) proceeds as follows at time t: with probability $1-\epsilon$, draw a pair (X_t, Y_t) from the distribution P_{z_t} where z_t is the attribute with largest empirical loss; and with probability ϵ draw (Z_t, X_t, Y_t) from the population distribution P_{XYZ}. If $\pi_Z(\cdot)$ denotes the marginal of the population distribution over $\mathcal{Z}$, i.e., $\pi_Z(z) = \sum_{x,y} P_{XYZ}(x,y,z)$; then the ϵ-`greedy` strategy can be equivalently described as follows: For any $t = 1, 2, \ldots$, do

- Draw a random variable $Q_t \sim$ `Bernoulli`(ϵ).
- If $Q_t = 1$, draw $Z_t \sim \pi_Z$. Else set $Z_t \in \arg\max_{z\in\mathcal{Z}} \hat{L}_t(z, \hat{f}_t)$.
- Draw a pair of samples from the distribution P_{Z_t}.

For simplicity of presentation, we assume that π_Z is the uniform distribution, i.e., $\pi_Z(z) = 1/m$ for all $z \in \mathcal{Z}$. However, our theoretical results can be easily generalized to the case of any π_Z which places non-zero mass on all $z \in \mathcal{Z}$.

C.1 Analysis of the ϵ-`greedy` Sampling Strategy

The ϵ-`greedy` strategy proposed in Abernethy et al. (2020) proceeds as follows: for any $t \geq 1$, it maintains a candidate classifier $\hat{f}_t$ along with its empirical loss on m validation sets corresponding to the attributes $z \in \mathcal{Z}$. At each round t, the algorithm selects an attribute z_t as follows: with probability ϵ, z_t is drawn from a fixed distribution over $\mathcal{Z}$, and with probability $1-\epsilon$, it is set to the distribution with the largest empirical loss. Having chosen z_t, the algorithm draws a pair of samples from P_{z_t} and updates the training and validation sets, as well as the classifier $\hat{f}_t$. This process is continued until the sampling budget is exhausted.

Abernethy et al. (2020) present two theoretical results on the performance of their ϵ-`greedy` strategy. The first one (Abernethy et al., 2020, Theorem 1), is an asymptotic consistency result derived under somewhat restrictive assumptions: $\mathcal{X} = \mathbb{R}$, $\mathcal{F}$ is the class of threshold classifiers, and the learner has access to an oracle which returns the exact loss, $L(z, f_{\pi_t})$, for every π_t. In their second result (Abernethy et al., 2020, Theorem 2), they analyze the greedy version (i.e., $\epsilon = 0$) of the algorithm with $|\mathcal{Z}| = 2$, and show that at time n, either the excess risk is of $\mathcal{O}\big(\max_{z\in\mathcal{Z}} \sqrt{2d_{VC}(\ell\circ\mathcal{F})\log(2/\delta)/N_{z,n}}\big)$, or the algorithm draws a sample from the attribute with the largest loss. However, due to the greedy nature of the algorithm analyzed, there are no guarantees that $N_{z,n} = \Omega(n)$, and thus, in the worst case the above excess risk bound is $\mathcal{O}(1)$.

We now show how the techniques we developed for the analysis of Algorithm 2 can be suitably employed to study the ϵ-`greedy` strategy under the same assumptions as in Theorem 1. In our results, we assume that the distribution according to which ϵ-`greedy` selects an attribute with probability ϵ at each round t is uniform. This is just for simplicity and all our results hold for any distribution which places non-zero mass on all $z \in \mathcal{Z}$.

We first present an intuitive result that says if the ϵ-`greedy` strategy is too *exploratory* (ϵ is large), the excess risk will not converge to zero. We present an illustration of this result using Instance I of the `SyntheticModel1` introduced in Section 2.2 in Figure 9 in Appendix C.

Proposition 2. *If the ϵ-`greedy` strategy is implemented with $\epsilon > m\pi_{\min} := |\mathcal{Z}| \min_{z\in\mathcal{Z}} \pi^*(z)$, then its excess risk is $\Omega(1)$ with probability at least $1-\delta$ for n large enough (see Equation 21 in Appendix C.2 for the precise condition on n).*

Proof Outline. The result follows from two observations: **(i)** For all t larger than a term $\tau_0(\delta, \epsilon, \pi_{\min})$, defined in (21), with probability at least $1-\delta$, for any $z \in \mathcal{Z}$, we must have $\pi_t(z) \geq \frac{\pi_{\min}+\epsilon/m}{2}$, and **(ii)** Suppose $\pi_{\min}$ is achieved by some attribute $z_{\min}$, i.e., $\pi^*(z_{\min}) = \pi_{\min}$. Then, the first statement implies that the excess risk of the ϵ-`greedy` algorithm is at least $\min_{\pi:\pi(z_{\min})\geq(\pi_{\min}+\epsilon/m)/2} L(z_{\min}, f_\pi) - M^*$, which is strictly greater than zero by Assumption 1. The detailed proof is reported in Appendix C.2. □

According to Proposition 2, for the excess risk to converge to zero, the ϵ-`greedy` strategy must be implemented with $\epsilon \leq \pi_{\min}/m$. We next derive an upper-bound on the excess risk of ϵ-`greedy`, similar to the one in Theorem 1 for Algorithm 2.

Theorem 2. *Let Assumptions 1-3 hold and ϵ-`greedy` implemented with $0 < \epsilon < m\pi_{\min}$. If the query budget n is sufficiently large (see Equations 23 and 24 in Appendix C.2 for the exact requirements), then for any $0 < \beta < \frac{m\pi_{\min}}{\epsilon} - 1$, with probability at least $1 - 2\delta$, we have*

$$\mathcal{R}_n(\epsilon\text{-}\texttt{greedy}) = \mathcal{O}\Big(\frac{|\mathcal{Z}|C}{q} \max_{z\in\mathcal{Z}} e_z(N_q)\Big), \tag{20}$$

where $q = \epsilon/m$, $N_q = qn\big(\pi_{\min} - q(1+\beta)\big)(1-\beta)$, and C is the parameter introduced in Assumption 3.

Remark 4. *The assumption on ϵ in Theorem 2 implies that $q = \epsilon/m < \pi_{\min}$, and as a result $qn\pi_{\min} < \pi_{\min}^2 n$. Given this and the monotonicity of the size of the confidence interval $e_z(N)$ w.r.t. N, we may conclude that the bound on the excess risk of Alg. 2 (Equation 5) is always tighter than the one for ϵ-`greedy` strategy (Equation 20). Note that for the classifiers with finite VC-dimension, the bound in* (20) *is of the same order in n as the one in* (5)*, but with a larger leading constant.*

C.2 Proof of Proposition 2

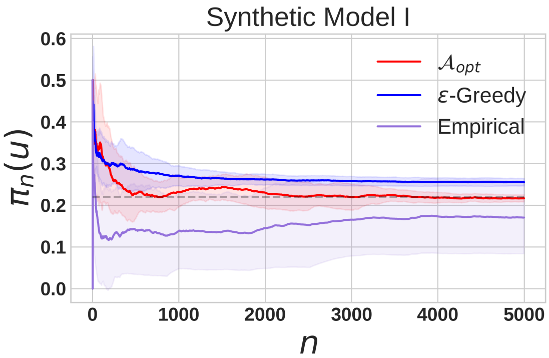

Figure 9: This figure shows an instance when the ϵ-`greedy` strategy is over-exploratory which results in the empirical mixture distribution π_t being strictly away from π^*, and thus resulting in $\Omega(1)$ excess risk even as n goes to infinity. For the ϵ-`greedy` strategy in this particular figure, we used $\epsilon = .5$ with $m = 2$ and $\pi_{\min} = \min_z \pi^*(z) \approx 0.23$. This provides a numerical demonstration of the statement of Proposition 2. Note that the $\mathcal{A}_{\texttt{opt}}$ strategy is more resilient to the wrong choice of the exploration parameter c_0: in this figure we used $c_0 = 1.0$ (much larger than the 0.1 value used in experiments) and the corresponding $\pi_n(u)$ for $\mathcal{A}_{\texttt{opt}}$ still eventually converges towards $\pi^*(u)$.

Fix any $z \in \mathcal{Z}$, and decompose $N_{z,t}$ into $N_{z,t}^{(0)} + N_{z,t}^{(1)}$ where, $N_{z,t}^{(1)} = \sum_{i=1}^t \mathbb{1}_{\{z_i=z\}}Q_i$ is the number of times up to round t the attribute z was queried due to the exploration step (i.e., $Q_t = 1$) of the ϵ-`greedy` algorithm.

Now, define $M_{z,0} = 0$ and $M_{z,t} = M_{z,t-1} + Q_t\mathbb{1}_{\{\tilde{z}_t=z\}} - \epsilon/m$. Due to the fact that Q_t and $\tilde{z}_t$ are independent, it is easy to check that $(M_{z,t})_{t\geq 0}$ forms a martingale sequence. For some given $\delta > 0$, introduce the notation $\delta_t = \frac{6\delta}{m\pi^2t^2}$ and $b_t = \sqrt{\frac{t\log(2/\delta_t)}{2}}$. We then have the following:

$$\begin{aligned}\sum_{t=1}^n P\left(M_{z,t} > b_t\right) &\leq \sum_{t=1}^n \left(\prod_{i=1}^t \mathbb{E}\left[\exp\left(\lambda Q_i\mathbb{1}_{\{\tilde{z}_i=z\}} - \epsilon/m\right)\big|(Q_j)_{j=1}^{i-1}\right]\right) e^{-\lambda b_t} \\ &\leq \sum_{t=1}^n e^{t\lambda^2/8-\lambda b_t} \overset{(a)}{=} \sum_{t=1}^n e^{-2b_t^2/t} \leq \sum_{t=1}^\infty \frac{\delta_t}{2} = \frac{\delta}{2m}.\end{aligned}$$

In the above display, **(a)** follows by setting $\lambda = 4b_t/t$. By repeating the same argument with the martingale sequence $\{-M_{z,t} : t \geq 0\}$, we get that $P\left(|M_{z,t}| \leq b_t, \forall t \geq 1,\ \forall z \in \mathcal{Z}\right) \geq 1 - \delta$. In other words, it implies that, with probability at least $1 - \delta$, we have the following:

$$N_{z,t}^{(1)} \geq \frac{t}{\epsilon} - b_t, \quad \forall\, t \geq 1,\ \forall z \in \mathcal{Z}.$$

This implies that $N_{z,t} \geq {}^{\epsilon}\!/m - b_t$ for all z, t. In particular, if $z_{\min}$ is the attribute such that $\pi^*(z_{\min}) = \min_{z \in \mathcal{Z}} \pi^*(z) = \pi_{\min}$, then this implies that $\hat{\pi}_t(z_{\min}) \geq \epsilon/m - b_t/t$ for all t, z. Using the fact that $\lim_{t \to \infty} b_t/t = 0$, we define $\tau_0 = \tau_0(\delta, \epsilon, \pi_{\min})$, as follows:

$$\tau_0(\delta, \epsilon, \pi_{\min}) \coloneqq \min \left\{ t \geq 1 : \sqrt{\frac{\log\left({}^{m\pi^2 t^2}\!/_{3\delta}\right)}{2t}} \leq \frac{\epsilon - m\pi_{\min}}{2m} \right\}. \tag{21}$$

This implies that with probability at least $1 - \delta$, $\pi_t(z_{\min}) \geq (\pi_{\min} + \epsilon/m)/2$ for all $t \geq \tau_0$. Hence the excess risk of the ϵ-`greedy` algorithm under these conditions is at least $\min_{\pi : \pi(z_{\min}) \geq (\pi_{\min} + \epsilon/m)/2} \mathbb{E}_{z_{\min}} \left[\ell\left(f_\pi, X, Y\right)\right] - M^*$, which is an $\Omega(1)$ term due to Assumption 1.

C.3 Proof of Theorem 2

First we partition the interval $[1, n]_{\mathbb{N}}$ into T_0 and T_1, where $\mathrm{T}_j = \{t : Q_t = j\}$ for $j = 0, 1$. In other words, T_0 denotes the times at which the ϵ-`greedy` strategy is *greedy* while T_1 denotes the times at which $Q_t = 1$ and the ϵ-`greedy` strategy is exploratory.

Then we introduce the following terms:

- Define $\mathcal{Z}_o$ and $\mathcal{Z}_u$ as in the proof of Theorem 1. That is, $\mathcal{Z}_o = \{z \in \mathcal{Z} : \pi_n(z) > \pi^*(z)\}$ and $\mathcal{Z}_u = \mathcal{Z} \setminus \mathcal{Z}_o$. We assume, as before, that $\mathcal{Z}_o \neq \emptyset$. Then, we define $t_0 \coloneqq \max\{t \in \mathrm{T}_0 : z_t \in \mathcal{Z}_o\}$, and denote the corresponding attribute (queried at time t_0) with z_0.
- Then, by appealing to Lemma 3, we note that there must exist a nonempty $\mathcal{Z}_o^{(1)} \subset \mathcal{Z}_u$ such that $\pi_n(z) > \pi_{t_0}(z)$ for all $z \in \mathcal{Z}_o^{(1)}$. Using this define $t_1 = \max\{t \in \mathrm{T}_0 : z_t \in \mathcal{Z}_o^{(1)}\}$ and use z_1 to denote the corresponding attribute queried at time t_1. We can proceed in this way to define $\{(t_j, z_j) : 2 \leq j \leq s\}$ for an appropriate $s \leq m - 1$ (such that $t_s = n$).

First, we define the $1 - \delta$ probability event Ω_3 as follows (see proof of Proposition 2 in Appendix C.2 for derivation):

$$\Omega_3 = \left\{ \left| N_{z,t}^{(1)} - \frac{\epsilon}{m} t \right| \leq \underbrace{\sqrt{\frac{t \log(m\pi^2 t^2 / 3\delta)}{2}}}_{\coloneqq b_t} \right\}. \tag{22}$$

For the rest of this proof, we will assume that the $(1 - 2\delta)$ probability event $\cap_{i=1}^3 \Omega_i$ holds, where Ω_1 and Ω_2 are the same uniform deviation results that were used in defining the UCB in (4).

Then we have the following:

Lemma 6. *At time t_0, we have $L(z, f_{\pi_{t_0}}) \leq M^* + \tilde{B}_0$ where $\tilde{B}_0 \coloneqq 4\left(1 + {}^{C}\!/_{q - \beta_{t_0}/t_0}\right) \max_{z \in \mathcal{Z}} e_{z, N_{z,t_0}}$ and $q = \epsilon/m$.*

Proof.

$$\begin{aligned}
L(z, f_{\pi_{t_0}}) &\leq \widehat{L}_{t_0}(z, f_{\pi_{t_0}}) + 2e_z(N_{z,t_0}) \\
&\leq \widehat{L}_{t_0}(z, \hat{f}_{t_0}) + 2e_z(N_{z,t_0}) + \frac{2C}{\pi_{t_0}(z)} \rho_{t_0} \\
&\leq \widehat{L}_{t_0}(z_0, \hat{f}_{t_0}) + 2e_z(N_{z,t_0}) + \frac{2C}{\pi_{t_0}(z)} \rho_{t_0} \\
&\leq L(z_0, f_{\pi_{t_0}}) + 2\left(e_z(N_{z,t_0}) + e_{z_0}(N_{z_0,t_0})\right) + 2C\rho_{t_0} \left(\frac{1}{\pi_{t_0}(z_0)} + \frac{1}{\pi_{t_0}(z)} \right) \\
&\leq L(z_0, f_{\pi_{t_0}}) + 4\left(1 + \frac{C}{\epsilon/m - b_{t_0}/t_0}\right) \max_{z \in \mathcal{Z}} e_z(N_{z,t_0}) \\
&= L(z_0, f_{\pi_{t_0}}) + \tilde{B}_0.
\end{aligned}$$

□

Lemma 7. *Suppose n is large enough to satisfy the conditions in* (24) *for some fixed* $0 < \beta < \pi_{\min}/q - 1$. *Then we have* $\tilde{B}_0 \leq 4\left(1 + {}^{C}/_{q(1-\beta)}\right) \max_{z \in \mathcal{Z}} e_z(N_q)$, *where* $N_q := nq\left(\pi_{\min} - q(1+\beta)\right)(1-\beta)$.

Due to the definitions of event Ω_3 and the time t_0, it must be the case that $N_{z_0,t_0} \geq N^{(0)}_{z_0,n} \geq \pi^*(z_0)n - N^{(1)}_{z_0,n} \geq \left(\pi^*(z_0) - \epsilon/m = -b_n/n\right) n$. Next, we assume that n is large enough, such that the following are satisfied for some $0 < \beta$:

- First, we assume that n is large enough to ensure that $b_n/n \leq \beta\epsilon/m$ for some $0 < \beta < m\pi_{\min}/\epsilon - 1$, i.e.,

$$n \geq \tilde{n_0}(\beta) := \min\left\{ t \geq 1 : \frac{\log t}{t} \leq \frac{\beta^2 \epsilon^2}{m^2} - \frac{1}{2}\log\left({}^{m\pi^2}/_{3\delta}\right) \right\}. \tag{23}$$

  This implies that $t_0 \geq n\left(\pi^*(z_0) - \epsilon/m(1+\beta)\right) \geq n\left(\pi_{\min} - \frac{\epsilon(1+\beta)}{m}\right)$.

- Next, we assume that $b_{t_0}/t_0 \leq \beta\epsilon/m$. A sufficient condition for this is

$$n\left(\pi_{\min} - \frac{\epsilon(1+\beta)}{m}\right) \;\geq\; \tilde{n}_0(\beta) \quad \Rightarrow \quad n \;\geq\; \frac{\tilde{n}_0(\beta)}{\pi_{\min} - {}^{\epsilon(1+\beta)}/_{m}}. \tag{24}$$

Then, with the notation $N_q = n\left(\pi_{\min} - q(1+\beta)\right)\left(q(1-\beta)\right)$, where $q = \epsilon/m$, we have the following:

$$\tilde{B}_0 = 4\left(1 + \frac{C}{q - b_{t_0}t_0}\right) \max_{z \in \mathcal{Z}} e_z(N_{z,t_0}) \leq 4\left(1 + \frac{C}{q(1-\beta)}\right) \max_{z \in \mathcal{Z}} e_z(N_q).$$

Now, proceeding in the same way as in Lemma 4 and Lemma 5, we can define the terms $\tilde{B}_j$ for $j \geq 1$ (analogous to the terms B_j introduced in Lemma 4) to show that with probability at least $1 - 2\delta$, the excess risk resulting from the ϵ-`greedy` strategy satisfies:

$$\max_{z \in \mathcal{Z}} L(z, f_{\pi_n}) - M^* \leq \sum_{j=0}^{s-1} \tilde{B}_j = \mathcal{O}\left(\frac{|\mathcal{Z}|C}{q(1-\beta)}\right) \max_{z \in \mathcal{Z}} e_z(N_q),$$

as required.

C.4 Comparison with $\mathcal{A}_{\texttt{opt}}$

The ϵ-`greedy` strategy differs from $\mathcal{A}_{\texttt{opt}}$ in two major ways:

1. The excess risk bound derived in Theorem 1 for $\mathcal{A}_{\texttt{opt}}$ is always tighter than the corresponding bound for ϵ-`greedy` strategy in Theorem 2. In particular, for the family of classifiers with finite VC dimension, both the algorithms achieve same convergence rates w.r.t. n, but the ϵ-`greedy` strategy has a larger leading constant.
2. From a practical point of view, the ϵ-`greedy` strategy is less robust to the choice of parameter ϵ as compared to the $\mathcal{A}_{\texttt{opt}}$ strategy. For instance, as shown in Figure 9, choosing a large value of ϵ may result in the mixture distribution (π_t) not converging to π^*, whereas even with much larger values of c_0, the π_t from $\mathcal{A}_{\texttt{opt}}$ algorithm still eventually converges to π^*.

D Proof of Lower Bound

In this section, we first formally state the lower bound result and then present its proof. We denote by $\mathcal{M}$ the class of `SyntheticModel1`s introduced in Definition 1. We also define the following class of problems:

Definition 3. Let $\mathcal{Q}$ denote the class of problems defined by the triplets $(\boldsymbol{\mu}, \mathcal{F}, \ell_{01})$, where $\boldsymbol{\mu} \in \mathcal{M}$ is an instance of the `SyntheticModel1`, $\mathcal{F}$ is the class of linear classifiers in two dimensions, and ℓ_{01} is the $0 - 1$ loss.

For the function class $\mathcal{F}$, we know that $e_z(N) = \mathcal{O}(\sqrt{\log(n/\delta)/N})$ for both $z \in \mathcal{Z} = \{u, v\}$, which implies that the expected excess risk achieved by both $\mathcal{A}_{\texttt{opt}}$ and ϵ-`greedy` strategies is of $\mathcal{O}\left(\sqrt{\log(n)/n}\right)$. We now prove that this convergence rate (in terms of n) for this class of problems.

Proposition 3. *Suppose $\mathcal{A}$ is any adaptive sampling scheme which is applied with a budget n to a problem $Q \in \mathcal{Q}$ introduced in Definition 3. Then, we have*

$$\max_{Q \in \mathcal{Q}} \mathbb{E}_Q \left[\mathcal{R}_n\left(\mathcal{A}\right)\right] = \Omega\left(1/\sqrt{n}\right). \tag{25}$$

To prove this proposition, we consider two problem instances in the class of problems, $\mathcal{Q}$, used in the statement of Proposition 3, denoted by Q_μ and Q_γ. The instance Q_μ has the synthetic model with mean vectors $\mu_{0u} = (-r, r), \mu_{1u} = (r, -r), \mu_{0v} = (-r', -r')$ and $\mu_{1v} = (r', r')$ for some $r' > r > 0$, and Q_γ has the mean vectors $\mu_{0u} = (-r', r'), \mu_{1u} = (r', -r'), \mu_{0v} = (-r, -r)$ and $\mu_{1v} = (r, r)$. For both these problem instances, implementing an adaptive sampling algorithm $\mathcal{A}$ with a budget n induces a probability measure on the space $(\mathcal{X} \times \mathcal{Y} \times \mathcal{Z})^n$. We use $\mathbb{P}_\mu$ and $\mathbb{P}_\gamma$ (resp. $\mathbb{E}_\mu$ and $\mathbb{E}_\gamma$) to denote the probability measures (resp. expectations) for the problem instances Q_μ and Q_γ respectively.

Then we have the following KL-divergence decomposition result.

Lemma 8. *For any event E, we have the following:*

$$\frac{n(r'-r)^2}{2} = D_{KL}\left(\mathbb{P}_\mu, \mathbb{P}_\gamma\right) \geq d_{KL}\left(\mathbb{P}_\mu(E), \mathbb{P}_\gamma(E)\right) \geq 2\left(\mathbb{P}_\mu(E) - \mathbb{P}_\gamma(E)\right)^2.$$

In the above display, $d_{KL}(a_1, a_2)$ for $a_1, a_2 \in [0, 1]$ denotes the KL-divergence between two Bernoulli random variables with parameters a_1 and a_2 respectively.

As a consequence of the above result, we have $|\mathbb{P}_\mu(E) - \mathbb{P}_\gamma(E)| \leq (r' - r)\sqrt{n/2}$.

Proof. The first inequality is a consequence of the data-processing inequality for KL-divergence (Polyanskiy and Wu, 2015, Corollary 2.2), while the second inequality follows from an application Pinsker's inequality (Polyanskiy and Wu, 2015, Theorem 6.5).

We now show the derivation of the first equality in the statement. Let $\mathcal{H}_t$ denote the history at the beginning of round t, i.e., $\mathcal{H}_t = (Z_1, X_1, Y_1, \ldots, Z_{t-1}, X_{t-1}, Y_{t-1})$. Then for any sequence of $(Z_1, X_1, Y_1, \ldots, Z_n, X_n, Y_n)$, we have

$$\mathbb{P}_\mu\left(Z_1, \ldots, Y_n\right) = \prod_{t=1}^{n} \mathcal{A}\left(Z_t | \mathcal{H}_{t-1}\right) Q_\mu\left(Y_t | Z_t\right) Q_\mu\left(X_t | Y_t, Z_t\right).$$

We can write a similar expression for $\mathbb{P}_\gamma$ as well. Now, proceeding as in (Lattimore and Szepesvári, 2020, Lemma 15.1), we get the following divergence decomposition result

$$\begin{aligned} D_{KL}\left(\mathbb{P}_\mu, \mathbb{P}_\gamma\right) &= \mathbb{E}_\mu\left[N_{u,n}\right] D_{KL}\left(Q_\mu(\cdot, \cdot | Z = u), Q_\gamma(\cdot, \cdot | Z = u)\right) + \mathbb{E}_\mu\left[N_{v,n}\right] D_{KL}\left(Q_\mu(\cdot, \cdot | Z = v), Q_\gamma(\cdot, \cdot | Z = v)\right) \\ &= \left(\mathbb{E}_\mu\left[N_{u,n}\right] + \mathbb{E}_\mu\left[N_{v,n}\right]\right) \frac{(r'-r)^2}{2} = \frac{n(r'-r)^2}{2}, \end{aligned}$$

where the last equality uses the expression for KL-divergence between two multi-variate Gaussian distributions. □

Every linear classifier $f \in \mathcal{F}$ can be equivalently parametrized by a normal vector $w \in \mathbb{R}^2$ such that $f(x) = \text{sign}(\langle w, x \rangle)$. We know that if $\pi(u) = 1$, then the optimal linear classifier is the one with normal vector $w_1 = (1, -1)$. That is $f^*(x) = \text{sign}(\langle w, x \rangle)$. Similarly, when $\pi(u) = 0$, the optimal linear classifier is the one with $w_0 = (1, 1)$, and it varies continuously from w_0 to w_1 as $\pi(u)$ increases from 0 to 1. Now, let a be the value of $\pi(u)$ at which the optimal linear classifier $f_\pi(x) = \text{sign}(\langle w, x \rangle)$ with $w = (1, 0)$. In the next lemma, we consider the event $E = \{\pi_n(u) > a\}$.

Lemma 9. *Suppose π_n denotes the mixture distribution returned by the algorithm $\mathcal{A}$ after n rounds when applied to a problem $Q \in \{Q_\mu, Q_\gamma\}$. Introduce the event $E = \{\pi_n(u) > a\}$ and assume that $r' < 2r$. Recall that $a \in (0, 1)$ is value of $\pi(u)$ such that the corresponding optimal linear classifier has the normal vector $(1, 0)$; that is, $f_\pi(x) = sign(\langle x, (1, 0) \rangle)$. Then we have the following:*

$$\mathbb{E}_\mu\left[\mathcal{R}_n(\mathcal{A})\right] \geq \frac{c_r}{2\sqrt{2}}(r'-r)\mathbb{P}_\mu(E^c) \quad \textit{and} \quad \mathbb{E}_\gamma\left[\mathcal{R}_n(\mathcal{A})\right] \geq \frac{c_r}{2\sqrt{2}}(r'-r)\mathbb{P}_\gamma(E),$$

where $c_r := \min_{x\in[r,2r]} \left|\Phi'\left(\frac{x}{\sqrt{2}}\right)\right|$ *and* $\Phi(\cdot)$ *denotes the cumulative distribution function (cdf) of a standard Normal random variable.*

Proof. We present the details only for the first inequality as the second inequality follows in an entirely analogous manner by replacing E^c with E.

$$\begin{aligned}\mathbb{E}_\mu\left[\mathcal{R}_n(\mathcal{A})\right] &= \mathbb{E}_\mu\left[\mathcal{R}_n(\mathcal{A})\mathbb{1}_{\{E\}}\right] + \mathbb{E}_\mu\left[\mathcal{R}_n(\mathcal{A})\mathbb{1}_{\{E^c\}}\right] \\ &\geq \mathbb{E}_\mu\left[\mathcal{R}_n(\mathcal{A})\mathbb{1}_{\{E^c\}}\right] \\ &\overset{(a)}{\geq} \left(\frac{\Phi(r/\sqrt{2}) - \Phi(r'/\sqrt{2})}{2}\right)\mathbb{P}_\mu(E^c) \\ &\geq \min_{x\in[r,2r]}\left|\Phi'(x/\sqrt{2})\right|\left(\frac{r'-r}{2\sqrt{2}}\right)\mathbb{P}_\mu\left(E^c\right) \\ &= \frac{c_r}{2\sqrt{2}}(r'-r)\mathbb{P}_\mu\left(E^c\right).\end{aligned}$$

The key observation in the proof which relies on the choice of $\mathcal{F}$ as the set of all linear classifiers, is in step **(a)** above. This step uses the fact that under the event E^c, when $\pi_n(u) \leq a$, the minimax loss must be at least greater than $\frac{\Phi(r/\sqrt{2})-\Phi(r'/\sqrt{2})}{2}$ (here $\Phi(\cdot)$ denotes the cdf of the standard normal random variable). This follows from the fact that under model Q_μ, the classifier w_μ^* which corresponds to the optimal mixture distribution π_μ^* must satisfy the condition that $\langle w_\mu^*, (1,0)\rangle \geq 0$; whereas under the event E^c the optimal classifier w_a^* satisfies $\langle w_a^*, (1,0)\rangle \leq 0$ (by definition of event E). □

The final result now follows by combining the results of Lemma 8 and Lemma 9. In particular, we have the following:

$$\begin{aligned}\max_{Q\in\mathcal{Q}}\mathbb{E}_Q\left[\mathcal{R}_n\left(\mathcal{A}\right)\right] \geq \max_{Q\in\{Q_\mu,Q_\gamma\}}\mathbb{E}_Q[\mathcal{R}_n\left(\mathcal{A}\right)] &\overset{\text{(a)}}{\geq} \frac{1}{2}\left(\mathbb{E}_\mu\left[\mathcal{R}_n\left(\mathcal{A}\right)\right] + \mathbb{E}_\gamma\left[\mathcal{R}_n\left(\mathcal{A}\right)\right]\right) \\ &\overset{\text{(b)}}{\geq} \frac{c_r}{4\sqrt{2}}(r'-r)\left(\mathbb{P}_\mu\left(E^c\right) + \mathbb{P}_\gamma\left(E\right)\right) \\ &\overset{\text{(c)}}{\geq} \frac{c_r}{4\sqrt{2}}(r'-r)\left(1 - \left|\mathbb{P}_\mu\left(E\right) - \mathbb{P}_\gamma\left(E\right)\right|\right) \\ &\overset{\text{(d)}}{\geq} \frac{c_r}{4\sqrt{2}}(r'-r)\left(1 - (r'-r)\sqrt{\frac{n}{2}}\right) \\ &\overset{\text{(e)}}{\geq} \frac{c_r}{16\sqrt{n}}.\end{aligned}$$

In the above display:
(a) uses the fact that average is smaller than maximum,
(b) lower bounds $\mathbb{E}_\mu\left[\mathcal{R}_n\left(\mathcal{A}\right)\right]$ and $\mathbb{E}_\gamma\left[\mathcal{R}_n\left(\mathcal{A}\right)\right]$ using the result of Lemma 9,
(c) uses the fact that $\mathbb{P}_\mu(E^c) + \mathbb{P}_\gamma(E) = 1 - \mathbb{P}_\mu(E^c) + \mathbb{P}_\gamma(E) \geq 1 - |\mathbb{P}_\mu(E^c) - \mathbb{P}_\gamma(E)|$,
(d) follows by using the bound $|\mathbb{P}_\mu(E^c) - \mathbb{P}_\gamma(E)| \leq (r'-r)\sqrt{n/2}$ derived in Lemma 8, and finally,
(e) follows by setting $r' = r + 1/\sqrt{2n}$.

D.1 Extending to $|\mathcal{Z}| > 2$

The same idea employed for the case of $m = |\mathcal{Z}| = 2$ can be generalized for larger even values of m. In particular, let $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0,1\}$ as before, and for some even $m > 2$ let $\mathcal{Z} = \{1,2,\ldots,m\}$. Furthermore, let $\mathcal{Z}_o = \{1,3,\ldots,m-1\}$ and $\mathcal{Z}_e = \{2,\ldots,m\}$ denote the odd and even numbered elements of $\mathcal{Z}$ (in fact any partition of $\mathcal{Z}$ into two sets of size $m/2$ each works). For any $i \in \mathcal{Z}$, we will use $\mathcal{Z}_i$ to denote $\mathcal{Z}_o$ if i is odd, and $\mathcal{Z}_e$ if i is even.

Now, consider $m+1$ problem instances for the above $(\mathcal{X},\mathcal{Y},\mathcal{Z})$, denoted by $Q_0,\ldots,Q_m$, and defined as follows:

- For Q_0, we have $P_{Y|Z}(Y = 1|Z = z) = 1/2$ for all $z \in \mathcal{Z}$ and $P_{X|YZ}(\cdot|Y = y, Z = z) \sim N(\mu_{yz}, I_2)$ where

$$\mu_{yz} = \begin{cases} (r, r), & \text{for } y = 0, z \in \mathcal{Z}_o \\ (-r, -r), & \text{for } y = 1, z \in \mathcal{Z}_o \\ (r, -r), & \text{for } y = 0, z \in \mathcal{Z}_e \\ (-r, r), & \text{for } y = 1, z \in \mathcal{Z}_e \end{cases} \tag{26}$$

- For all other $z \geq 1$, the problem Q_z is exactly the same as Q_0 except for $P_{X|YZ}(\cdot|Y = y, Z = z)$ for $y \in \{0, 1\}$. In particular, $\mu_{oz} = (r', r')$, $\mu_{1z} = (-r', -r')$ if $z \in \mathcal{Z}_o$ and $\mu_{0z} = (r', -r')$ and $\mu_{1z} = (-r', r')$ if $z \in \mathcal{Z}_e$.

Now, as in the $|\mathcal{Z}| = 2$ case, for any $i \in \mathcal{Z}$, define $a = \sum_{z \in \mathcal{Z}_i} \pi(z) : w^*_\pi = (1, 0)$, and note that since $r' > r$, we must have $a > 1/2$. Introduce the notation E_i to denote the event $\{\sum_{z \in \mathcal{Z}_i} \pi_n(z) > a\}$, where π_n denotes the empirical mixture distribution constructed by a given algorithm $\mathcal{A}$ after n samples. Recall that for any $i \in \mathcal{Z}$, the set $\mathcal{Z}_i$ is equal to $\mathcal{Z}_o$ or $\mathcal{Z}_e$ depending on whether i is even or odd.

Then, proceeding as before, we have

$$\begin{aligned} \max_{0 \leq i \leq m} \mathbb{E}_{Q_i}\left[\mathcal{R}_n(\mathcal{A})\right] \geq \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}_{Q_i}\left[\mathcal{R}_n\left(\mathcal{A}\right)\right] &\overset{(i)}{\geq} \frac{c_r(r' - r)}{2\sqrt{2}} \frac{1}{m} \sum_{i=1}^{m} \mathbb{P}_{Q_i}\left(E_i^c\right) \\ &\geq \frac{c_r(r' - r)}{2\sqrt{2}} \frac{1}{m} \sum_{i=1}^{m} \left(1 - \mathbb{P}_{Q_i}\left(E_i\right)\right) \\ &\overset{(ii)}{\geq} \frac{c_r(r' - r)}{2\sqrt{2}} \frac{1}{m} \sum_{i=1}^{m} \left(1 - \mathbb{P}_{Q_0}\left(E_i\right) - d_{TV}\left(\mathbb{P}_{Q_i}, \mathbb{P}_{Q_0}\right)\right) \\ &\overset{(iii)}{\geq} \frac{c_r(r' - r)}{2\sqrt{2}} \left(1 - \frac{1}{m}\left(\frac{m}{2} + \sum_{i=1}^{m} (r' - r) \sqrt{\frac{\mathbb{E}_{Q_0}[N_{i,n}]}{2}}\right)\right) \\ &\overset{(iv)}{\geq} \frac{c_r(r' - r)}{2\sqrt{2}} \left(\frac{1}{2} - \frac{1}{m}\left((r' - r)\sqrt{\frac{nm}{2}}\right)\right) \\ &= \frac{c_r(r' - r)}{2\sqrt{2}} \left(\frac{1}{2} - (r' - r)\sqrt{\frac{n}{2m}}\right) \\ &\overset{(v)}{=} \Omega\left(\sqrt{\frac{m}{n}}\right). \end{aligned}$$

In the above display,
(i) follows from the definition of event E_i,
(ii) uses the fact that $|\mathbb{P}_{Q_0}(E_i) - \mathbb{P}_{Q_i}(E_i)| \leq d_{TV}(\mathbb{P}_{Q_0}, \mathbb{P}_{Q_i})$,
(iii) uses Pinsker's inequality to bound the total-variation distance,
(iv) uses Cauchy-Schwarz inequality along with the fact that $\sum_{i=1}^{m} N_{i,n} = n$,
(v) follows by setting $r' = r + \sqrt{m/2n}$.

E Details of Experiments

E.1 Synthetic Datasets

Data. We used the two synthetic models introduced in Section 2.2 for generating the training set. Here we provide a formal definition for `SyntheticModel2`, an instance of which is illustrated in Figure 7.

Definition 4 (`SyntheticModel2`)**.** Set $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0, 1\}$, and $\mathcal{Z} = \{u, v\}$ and fix an angle θ and a constant $a \in \mathbb{R}$. Then $P_{X|Y=y,Z=u} = \mathcal{U}(c_y, c_y + a)$. That is, within attribute $Z = u$ and for either label $Y = y$, X is uniformly distributed along one dimension. To get the conditional distributions for $Z = v$ we draw from mixtures of three Gaussians: $\widetilde{P}_{X=x|Y=y,Z=v} = \sum_{i=1}^{3} \rho_{i,y} P_{i,y}(x)$ where each $P_{i,y} = \mathcal{N}(\mu_{i,y}, \Sigma_{i,y})$ and $\rho_{i,y}$ are non-negative and sum to one over i. The resulting sample is then uniformly rotated counterclockwise by θ in the plane.

To elaborate on the intuition provided for this model in the main text, our goal here is to design a problem where one attribute, here $Z = u$, has a low maximum expected accuracy, for a linear classifier, that can be attained with relatively few samples. Here, any linear classifier will misclassify at least half of all points in the overlap of the intervals $(c_0, c_0 + a)$ and $(c_1, c_1 + a)$. And any linear classifier which intercepts this overlap will attain the maximal possible expected accuracy on $Z = u$. Then for $Z = v$ we choose the $\rho_{i,y}$ and covariances such that most mass is contained in easily separable clusters, so that a classifier can attain high accuracy on $Z = v$ with few samples. But, the sparser clusters are chosen to have some overlap between $Y = 1$ and $Y = 0$. The overlap and sparseness require a learner draw more samples from $Z = v$ to accurately learn the optimal boundary.

Thus $\mathcal{A}_{\texttt{opt}}$ and `Greedy` will continually sample from $Z = u$, as any linear classifier will have relatively low accuracy on it, despite the fact that additional samples do not improve performance on this attribute. In contrast, $\mathcal{A}_{\texttt{opt}}+$ identifies this stagnation in performance and samples from $Z = v$, over time drawing sufficient samples from the sparse clusters to maximize accuracy on both attributes.

Classifier. We set $\mathcal{F}$ to the set of logistic regression classifiers. More specifically we used the `LogisticRegression` implementation in the `scikit-learn` package.

Details of experiments. For the experiments on `SyntheticModel1`, pictured in Figure 2, for each trial we set $n = 1000$ and tracked the $\pi_t(u)$ values for the three algorithms. We repeated the experiment for 100 trials and plot the resulting mean $\pi_t(u)$ values in the curves, along with the one-standard-deviation regions.

For experiments on `SyntheticModel2`, pictured in Figure 6a and Figure 6b, we used the heuristic algorithm, $\mathcal{A}_{\texttt{opt}}+$, with $c_1 = 0.1$ and the Mann-Kendall statistic tracking the accuracy after the previous 20 sample draws for each attribute. We ran 100 trials for each scheme and report the mean accuracy over both attributes, along with one-standard-deviation regions.

E.2 Real Datasets

We used the following datasets:

- *UTKFace* dataset. This dataset consists of large number of face images with annotations of age, gender and ethnicity. We used $Y = \{$`Male, Female`$\}$ and set $\mathcal{Z} = \{$`White, Black, Asian, Indian, Other`$\}$.
- *FashionMNIST* dataset. This dataset consists of 10 different classes, which were paired off to get five different binary classification tasks: `{(Tshirt, Shirt), (Trousers, Dress), (Pullover, Coat), (Sandals, Bag), (Sneakers, AnkleBoots)}`
- *Cifar10* dataset. This dataset also consists of 10 different classes, which were paired off as follows: `{ (airplane, ship), (automobile, truck), (bird, cat), (deer, dog), (frog, horse)}`.
- *Adult* dataset. This is a low dimensional, binary classification problem where the inputs are a variety of demographic data about an individual and the task is to predict whether the individual makes more or less than \$50,000/year.
- *German* dataset. This is another low dimensional, binary classification problem where the inputs are demographic and financial information about an individual and the task is to categorize them as low or high risk for defaulting on a loan.

Data Transforms and Augmentation. We used the following pre-processing operations for the different datasets:

- *UTKFace.* For this dataset, we first resized the dataset to size $3 \times 50 \times 50$ from the original $3 \times 200 \times 200$. Then we also applied random horizontal flip with probability 0.5. For training we used the images from the file `UTKFace.tar.gz`, while for testing we used the file `crop_part1.tar.gz`. (Both of these files can be found at this link shared by the owner of the dataset).

- *FashionMNIST.* We only employed the normalization transform in this case, and used the default training and test splits.
- *Cifar10.* We employed a random crop transform (to size $3 \times 28 \times 28$ with padding 1), a random horizontal flip transform (with probability 0.5) and a normalization transform, and used the default training and test split.
- *Adult.* For the categorical inputs we used a one-hot encoding and normalized all numerical inputs to $[0, 1]$. We used the provided training/test split.
- *German* We used the same pre-processing as for the *Adult* dataset here. As described in the main paper, this dataset only provides 1000 examples and does not have a canonical train/test split. Moreover, results were very sensitive to the choice of split. So for every experiment we generated a new, random 70/30 training/test split and report results averaged over 500 such trials.

Details of Classifiers. We implemented the CNN using Pytorch. The CNN used for both *Cifar10* and *UTKFace* had the same architecture consisting of:

- First `Conv2d` layer with 32 output channels, kernel size 3, padding = 1, followed by `ReLU` followed by a `MaxPool2d` layer with kernel size 2 and stride 2.
- Second `Conv2d` layer with 64 output channels, kernel size 3, followed by `ReLU` followed by `MaxPool2d` with kernel size 2.
- The two `Conv2d` layers were followed by 3 fully connected layers with output sizes 600, 120 and 2 respectively.
- For training the neural network we used the Adam optimizer with $\texttt{lr} = 0.001$.

The CNN used for *FashionMNIST* was identical, except with the second convolutional layer omitted, due to the simpler nature of the dataset.

For both *Adult* and *German* datasets we again used the `LogisticRegression` implementation in the `scikit-learn` package.

Computing Infrastructure used. The image dataset experiments were run on Google Colab using the free GPU instances and on a shared computing cluster which provides a variety of different GPUs. So we cannot provide the exact details of the GPUs we used, but all experiments in the paper can comfortably run in less than day on a GTX 1080 TI.

E.3 Additional Experimental Results

Here we present results for experiments on additional datasets, these are all analogous to results presented in the main paper on other datasets.

First we have the *German* dataset, which is from the UCI Repository (Dua and Graff, 2017) and qualitatively similar to the *Adult* dataset. We again use a LR classifier and the exact implementation of each algorithm. We set $\mathcal{Z}$ to be male or female. The results, in Figure 10a, show, on average, an advantage for $\mathcal{A}_{\texttt{opt}}$ over both `Uniform` and `Greedy` schemes. But this dataset contains only 1000 examples in total. We generate a 70/30 training/test split, but find that our results strongly depend on

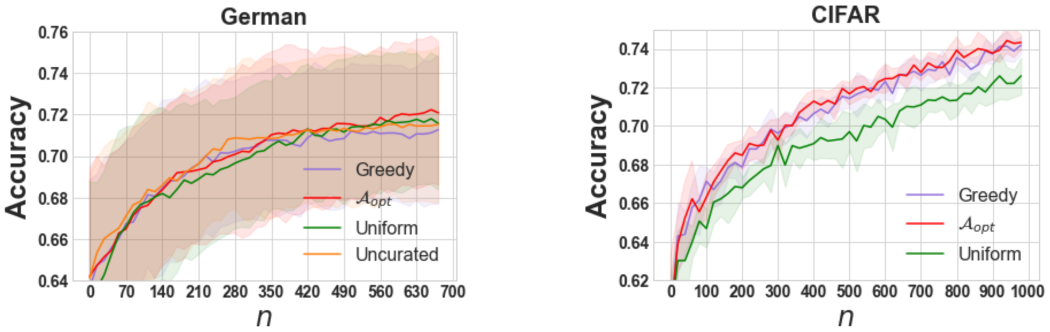

Figure 10: Left: Minimum test accuracy over all attributes for *German* dataset as a function of the sampling budget n, averaged over 500 trials. Right: Minimax error as a function of training round for *Cifar10*.

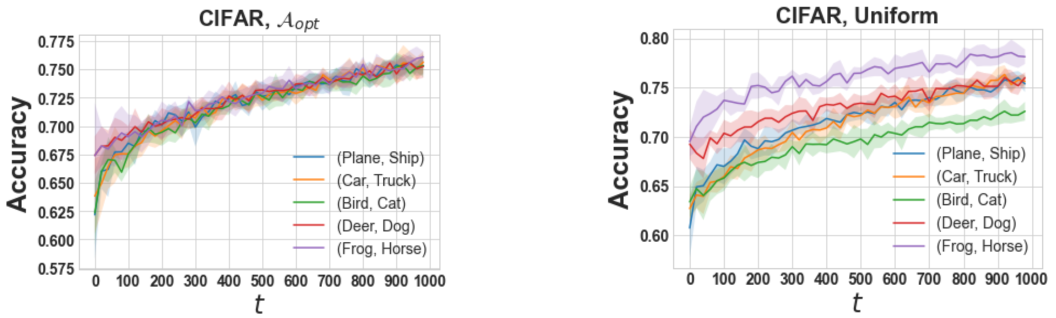

Figure 11: Test accuracy for each attribute in *Cifar10* as a function of the time step, t, for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` sampling schemes, averaged over 10 trials.

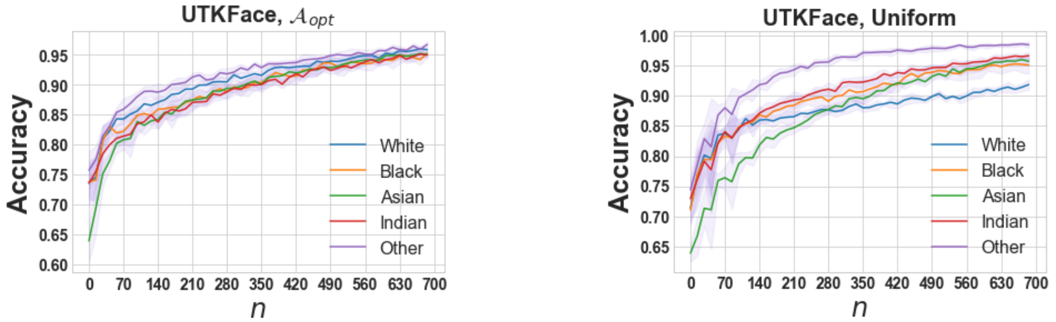

Figure 12: Test accuracy for each attribute in *UTKFace* as a function of the time step, t, for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` sampling schemes, averaged over 10 trials.

this split. To mitigate this, we run 500 trials and generate a new random split for each, but still find very high variance in our results, indicated by the shaded region in the figure.

Figure 10b shows the minimax error on *Cifar10*. This again demonstrates a significant improvement in worst case accuracy for the adaptive schemes over the `Uniform` scheme, which is equivalent to `Uncurated` for this dataset. Figures 11 and 12 show accuracy across all attributes as a function of training round for *Cifar10* and *UTKFace* for both $\mathcal{A}_{\texttt{opt}}$ and `Uniform` schemes, as in the analogous results for *FashionMNIST* this demonstrates the utility of $\mathcal{A}_{\texttt{opt}}$ for improving fairness between groups and lends credence to our assumption of the existence of an equalizing sampling distribution for real world.

Finally Table 1 summarizes the final test set accuracy, with standard deviation ranges, for each dataset and sampling scheme at the end of their respective training periods. As in other results, the adaptive schemes show an advantage over `Uniform` in all cases, and the efficacy of the `Uncurated` scheme depends on the nature of the dataset and chosen attributes.

Dataset	$\mathcal{A}_{opt}$	`Greedy`	`Uniform`	`Uncurated`
UTKFace	0.946 ± 0.003	0.946 ± 0.012	0.919 ± 0.008	0.933 ± 0.007
FashionMNIST	0.936 ± 0.004	0.924 ± 0.010	0.893 ± 0.002	0.893 ± 0.002
Cifar10	0.743 ± 0.004	0.742 ± 0.005	0.726 ± 0.010	0.726 ± 0.010
Adult	0.801 ± 0.001	0.800 ± 0.002	0.798 ± 0.002	0.797 ± 0.003
German	0.721 ± 0.035	0.713 ± 0.036	0.716 ± 0.032	0.715 ± 0.038

Table 1: Minimum test accuracy over different attributes achieved by the classifiers returned by the 4 sampling schemes.